ARL-MR-1068 ● Oct 2023

Terrain and Urban Data Preprocessing System for the Atmospheric Boundary Layer Environment-Lattice Boltzmann Model

by Leelinda P Dawson and Yansen Wang

NOTICES

Disclaimers

ARL-MR-1068 ● Oct 2023

Terrain and Urban Data Preprocessing System for the Atmospheric Boundary Layer Environment-Lattice Boltzmann Model

Leelinda P Dawson and Yansen Wang
DEVCOM Army Research Laboratory

REPORT DOCUMENTATION PAGE

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED: START DATE	3. DATES COVERED: END DATE
October 2023	Memorandum Report	1 October 2022	1 September 2023

4. TITLE AND SUBTITLE
Terrain and Urban Data Preprocessing System for the Atmospheric Boundary Layer Environment-Lattice Boltzmann Model

5a. CONTRACT NUMBER	5b. GRANT NUMBER	5c. PROGRAM ELEMENT NUMBER
5d. PROJECT NUMBER	**5e. TASK NUMBER**	**5f. WORK UNIT NUMBER**

6. AUTHOR(S)
Leelinda P Dawson and Yansen Wang

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
DEVCOM Army Research Laboratory ATTN: FCDD-RLA-NA Adelphi, MD 20783-1138	ARL-MR-1086

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited.

13. SUPPLEMENTARY NOTES
ORCID IDs: Leelinda P Dawson, 0000-0003-4209-8459; Yansen Wang, 0000-0003-1273-4290

14. ABSTRACT
The Atmospheric Boundary Layer Environment-Lattice Boltzmann Model (ABLE-LBM) is a next-generation microscale atmospheric model that has been developed by the US Army Combat Capabilities Development Command Army Research Laboratory to simulate and predict the atmospheric boundary layer turbulent wind, temperature, and other scalar transports over complex terrain and urban areas. The model requires grid-rasterized data of terrain elevation, urban building, and vegetation for the surface boundary information. This report details the successful proof of concept of using Node.js, an open-source JavaScript runtime environment, as a solution for performing the required preprocessing tasks of ABLE-LBM and in the development of a Google Geographical Information System (GIS)-based graphical user interface (GUI) web application for ABLE-LBM on a localhost server. The ABLE-LBM preprocessing tasks for terrain and building data generation were performed successfully using the ABLE-LBM Node.js GUI web application. This novel system takes advantage of enormous resources in map and image data retrieving and/or handling, overlaying, and many other GIS features that the Google Maps/Earth platform has to offer for improving the system's functionality and ease of use. The application is also cross-platform, which makes it more versatile for users.

15. SUBJECT TERMS
microscale atmospheric boundary layer modeling; terrain and urban data preprocessing; GIS application for atmospheric modeling; Node.js application; Network, Cyber, and Computational Sciences

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	C. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UU	17

19a. NAME OF RESPONSIBLE PERSON	19b. PHONE NUMBER *(Include area code)*
Leelinda P Dawson	(301) 394-5636

STANDARD FORM 298 (REV. 5/2020)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures

1. Introduction

The Atmospheric Boundary Layer Environment-Lattice Boltzmann Model (ABLE-LBM)[1,2] is a next-generation microscale atmospheric model that has been developed by the US Army Combat Capabilities Development Command (DEVCOM) Army Research Laboratory (ARL). It simulates turbulent atmospheric boundary layer wind, temperature, moisture, and aerosol transports over complex terrain and urban areas. The model results can be applied to unmanned aircraft system flight control, acoustic/optical wave propagations, and transport and dispersion of materials in the atmospheric boundary layer.

Our legacy model, the Three-Dimensional Wind Field (3DWF),[3,4] was also developed by ARL and simulates atmospheric boundary flows. Subsequently, a Google Geographical Information System (GIS)-based graphical user interface (GUI) application was designed for 3DWF so the user can easily perform their preprocessing tasks, run the model, and visualize its 3-D output on a Windows desktop computer.[5] The 3DWF application was commonly used by ARL until further advancements in the development of the next-generation microscale model, ABLE-LBM, was achieved over the past few years. In addition, the 3DWF application used HyperText Markup Language (HTML), JavaScript, and ActiveX control commands via a Microsoft Internet Explorer (IE) web browser. Since both IE and ActiveX control commands were recently discontinued and stopped being supported by Microsoft, there was an immediate requirement to develop a Google GIS-based GUI web application for ABLE-LBM that provides the same or better functionality as the legacy 3DWF web application.

This report details the successful proof of concept of using Node.js,[6] an open-source JavaScript runtime environment, as a solution for performing the required preprocessing tasks of ABLE-LBM and in the development of a Google GIS-based GUI web application for ABLE-LBM on a localhost server. As a result, the user can seamlessly run the model and view its 3-D atmospheric impacts visualization output on a Windows or Linux computer via Microsoft Edge or Google Chrome web browsers.

2. Node.js Approach

Node.js is an open-source JavaScript runtime environment that can run on multiple computer platforms such as Windows and Linux and all major web browsers such as Microsoft Edge and Google Chrome on the localhost server. Node.js generates dynamic page content and collects HTML form data. It can also securely create, read, write, and execute files on the server and store them on the local computer.

Furthermore, the approach of using Node.js as a solution for the ABLE-LBM web application was a good choice, especially considering that the plan was for the ABLE-LBM web application to run on multiple web browsers and platforms. The legacy 3DWF GUI application did not have the capability to run on a Windows platform via IE. Furthermore, the functionality to read, write, and execute local data files was an important feature and requirement for the ABLE-LBM web application.

Node.js provides a set of various built-in modules that can be easily used without further installation via the Node Package Manager (NPM). The Node.js modules that were used for the ABLE-LBM web application were Express, Embedded JavaScript (EJS), and Extensible Markup Language-JavaScript (XML-JS). Express is the minimal Node.js web application framework that provides a broad set of features for building the web application. EJS is a simple templating engine that generates a HTML template with minimal code to produce dynamic HTML pages to the user. XML-JS is an application framework that converts XML data to JavaScript (or vice versa). These three built-in modules helped to accelerate the development of the ABLE-LBM Node.js web application.

The preliminary GUI for the ABLE-LBM web application was developed in HTML and JavaScript using Node.js that acts as the middleware software on the localhost server. In addition, the ABLE-LBM web application uses Google Maps for GIS purposes, so the user can quickly identify their area of interest using images and geodetic coordinates during the ABLE-LBM preprocessing stage. Both the hostname and port number of the localhost server is set statically to localhost and 3000, respectively, on Windows platforms (i.e., http://localhost:3000). However, the assignment of the port number for the localhost server is dynamic on Linux platforms. In other words, the port number for the localhost server (i.e., http://localhost:<port_num>, where <port_num> represents the port number) can vary with each new user instance of the Node.js web application on Linux platforms. Therefore, an additional method was created in the ABLE-LBM Node.js web application to resolve these differences with the localhost server based on the user's computer platform. To start the localhost server using the ABLE-LBM Node.js software package, the user must run "node ABLE-LBM.js" on the command prompt or simply run the script provided for their computer platform. Then, the user can enter the localhost web address displayed in the command prompt from running the localhost server in their web browser to start the ABLE-LBM Node.js GUI web application. The overall execution and data flow of the Node.js implementation for the ABLE-LBM web application is shown in Fig. 1.

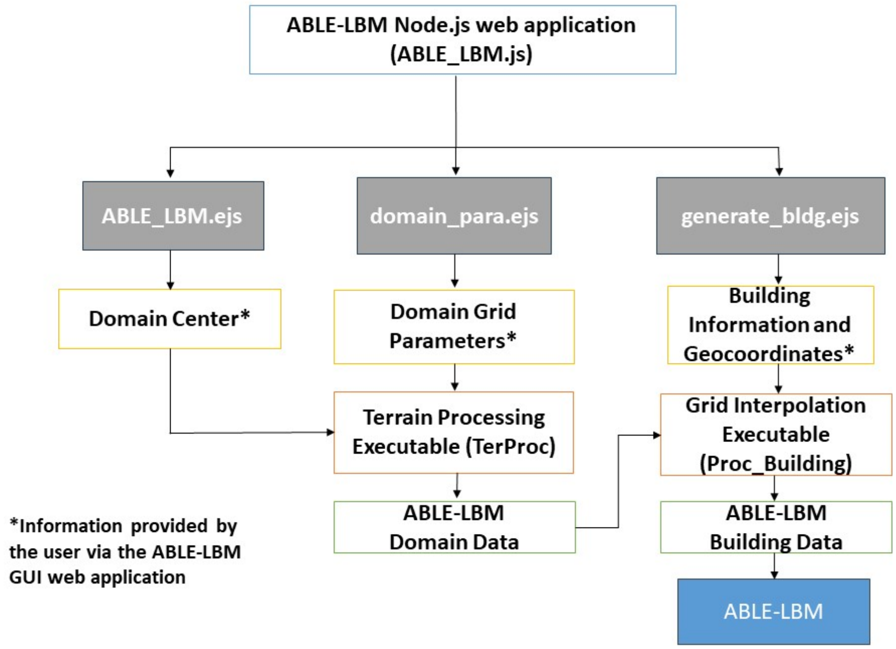

Fig. 1 Flowchart of ABLE-LBM Node.js implementation

The ABLE-LBM Node.js web application retrieves the required information from the user on the GUI. On the initial application startup, the user can manually enter the domain center that represents the latitude and longitude points of his or her area of interest, or mouse click on the Google Map located on the application to select the domain center, as shown in Fig. 2. Then, the application stores the latitude and longitude values of the domain center into a local file on the computer so it can be used for ABLE-LBM terrain data processing (as described in Section 3.1). Additional information about the buildings in the area of interest are also obtained from the user on the GUI, and these data are stored locally to perform grid interpolations for ABLE-LBM building data generation (as described in Section 3.2).

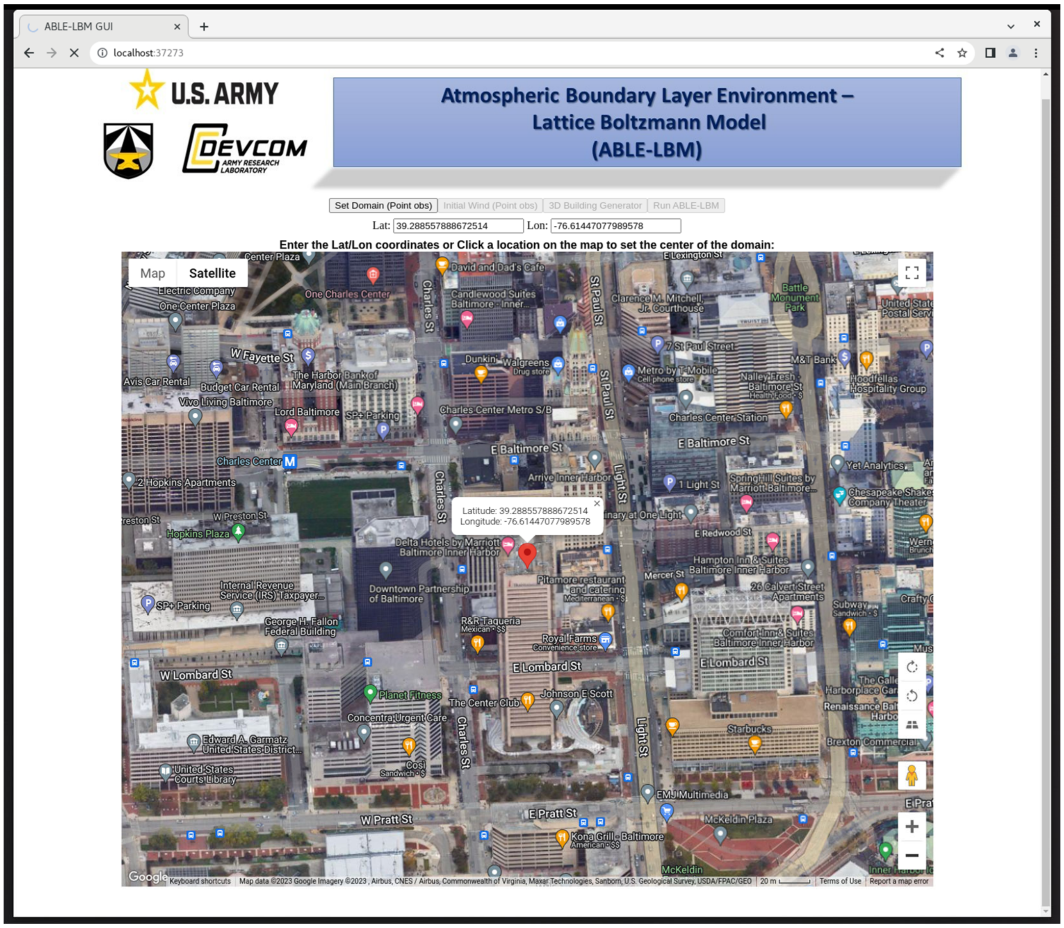

Fig. 2 Set ABLE-LBM domain center

3. ABLE-LBM Ground Surface Data Preprocessing

There are multiple preprocessing tasks that are required to be completed before running ABLE-LBM. The preprocessed data of these tasks provide surface boundary information for all the surface physical processes in the model, such as mechanical forces on the atmospheric boundary layer flows. This section of the report covers the two data preprocessing tasks required to run the model that is associated with (1) terrain data processing (Section 3.1) and (2) building data generation and corner coordinate transformation (Section 3.2).

3.1 Terrain Data Processing

The preprocessing task of terrain elevation involves collecting information about the domain center and grid parameters (i.e., grid length in meters and grid numbers) from the user on the GUI. The terrain processing interpolates terrain elevation data according to the model domain parameter inputs (number of grids and resolution of the model grid) from the user. The terrain elevation data are from the NASA Shuttle

Radar Topography Mission (SRTM),[7] which covers almost the entire land area of the earth globe from 56 south to 60 north latitudes. We chose the 1″ digital elevation data (~30-m resolution horizontal resolution) for its high resolution, quality, and nearly complete global coverage. The data tiles can be downloaded from the US Geological Survey (USGS) web site. Each tile of elevation data contains an area of 1° × 1° and is stored in the band interleaved by line (BIL) format. The BIL format is a binary raster format with an accompanying header file that describes the layout and formatting of the file. The name of the SRTM data file tiles is computed from the GUI, according to the ABLE-LBM domain parameters, so that the user can easily find and download the necessary data tiles. The data are then ingested into a terrain processing executable program called TerProc that creates the gridded binary terrain elevation data file for a model domain according to the domain grid numbers and grid resolution specification. This geointerpolation program produces a data file called ht_lat_long_ABLE_LBM_domain.dat, which includes the elevation, latitude, and longitude of each grid in the computational domain. This data file is used in both the subsequent data processing and visualization of model results on the GIS. When the terrain elevation data processing is completed, the GUI will display the four latitude and longitude corners of the specified ABLE-LBM domain, as shown in Fig. 3.

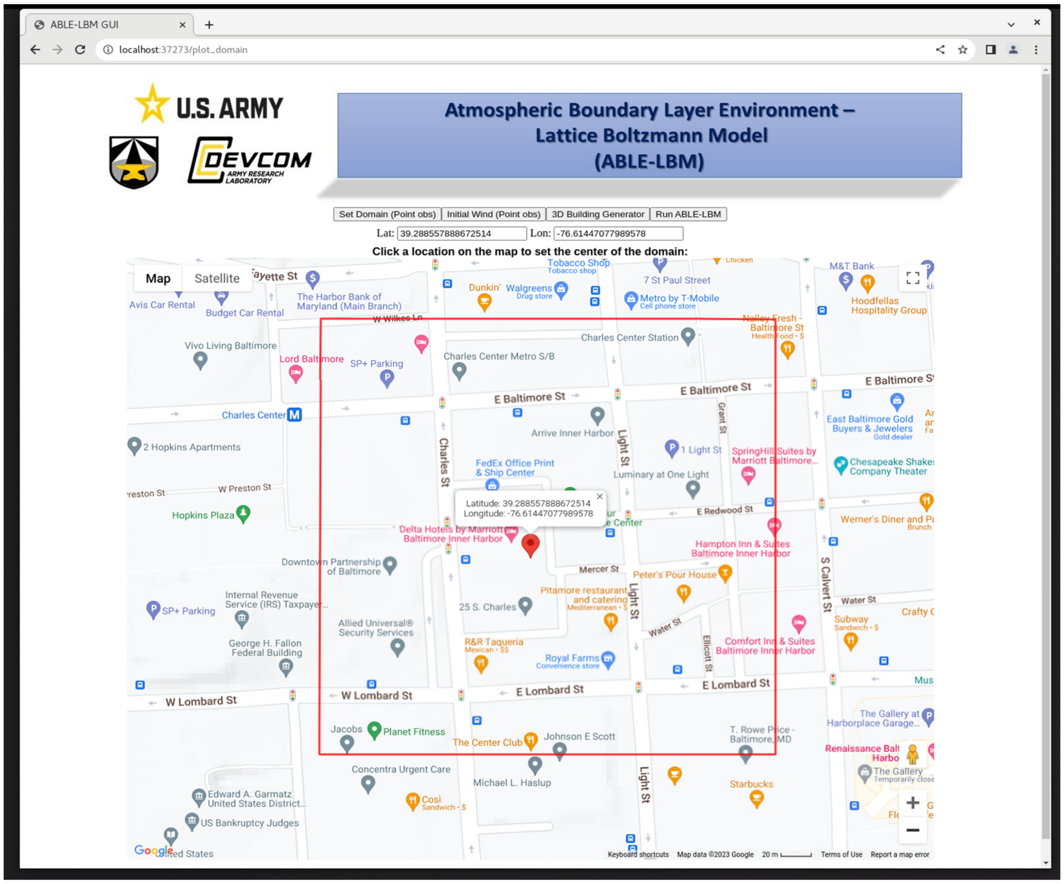

Fig. 3 ABLE-LBM domain plot after terrain data processing

Terrain data processing not only involves the interpolation the terrain elevation data, but it also goes through several geodetic coordinate transformations. The SRTM data were organized in the World Geodetic System 84 (WGS84) geodetic coordinate system. The ABLE-LBM is a microscale atmospheric boundary layer model. Because the model has a fine-resolution grid (i.e., meter to several meters) and the model domain usually covers no more than a 5- × 5-km surface area on the earth, the model uses the Cartesian coordinate with little distortion. The WGS84 coordinated data are transformed to a local East, North, and UP (ENU) Cartesian system. Furthermore, the interpolation of the SRTM terrain elevation data for every grid point of ABLE-LBM uses the standard bilinear interpolation method. A detailed description about the coordinate transformation and interpolation is described in Wang et al.[3]

3.2 Building Data Generation and Corner Coordinate Transformation

The preprocessing task of grid interpolation for building data generation involves retrieving information about the buildings in the selected domain from the user on the GUI. Building information includes the number of buildings of interest in the domain and for each building, the geocoordinates (or the latitude and longitude points) of their building corners, and the number of stories. The geocoordinates are retrieved by the user mouse clicking on a Google Map to plot clockwise and draw polygons of each building's roof on the GUI.

All the input data into ABLE-LBM must be in the model ENU coordinate system. Once all the ABLE-LBM building geocoordinate data are collected, then these data are ingested into the grid interpolation executable called Proc_Building. This is used to generate building data in ENU coordinate that include the domain x and y coordinates for each of the building corners in the ABLE-LBM domain.

During the building data process, the terrain data file called ht_lat_long_ABLE_LBM_domain.dat (as discussed in Section 3.1) is used for the coordinate transformation and interpolation. Using the building corner geocoordinates and the terrain data file, the closest four grid points in model domain in ENU coordinate can be found. The building corner points' exact x and y values (in meters) in ENU coordinates are interpolated from the four nearest grid points using the standard bilinear interpolation. Each grid building corner in the geocoordinate is searched and interpolated and then stored in a data file to build data input for the ABLE-LBM.

4. Conclusion and Future Work

ABLE-LBM, developed by ARL, is the next-generation atmospheric microscale model that simulates and predicts the atmospheric boundary layer turbulent wind, temperature, and other scalar transports over complex terrain and urban areas. Node.js provides a good solution to perform the preprocessing tasks needed for ABLE-LBM and serves as the middleware software for the ABLE-LBM GUI web application. In addition, Node.js allows the ABLE-LBM application to run on multiple computer platforms and web browsers, which helps to make the application more versatile for users. Currently, the ABLE-LBM Node.js GUI web application retrieves various domain information from the user's specified area of interest to perform the following preprocessing tasks: (1) terrain data processing and (2) building data generation and corner coordinate transformation. Then, the preprocessed data can be ingested into ABLE-LBM to produce ground surface mechanical and thermal forces to the atmospheric boundary layer flows.

The future work of the ABLE-LBM Node.js GUI web application involves the continuation of developing and improving the application. The next major step is to develop and integrate automated methods that can run ABLE-LBM after the final preprocessing stage, which involves the generation of building data and corner coordinate transformation (Fig. 1). The subsequent step is to develop and integrate automated visualization methods for the ABLE-LBM GUI web application, so that the 3-D atmospheric variable fields output from ABLE-LBM can be displayed on Google Maps or Google Earth and are easily understood by the user for their area of interest. Furthermore, the ABLE-LBM Node.js web application currently works for urban terrain with buildings. Future work also includes incorporating forest canopy that can also affect the turbulent flows and associated transports of mass and energy in the atmospheric boundary layer. At that time, atmospheric environment information can be provided to the user through the ABLE-LBM GUI web application for multiple terrain types.

5. References

1. Wang Y, Decker J, Pardyjak E. Large-eddy simulations of turbulent flows around buildings using a lattice Boltzmann model. J Appl Meteorol Climatol. 2020;59:885–899. doi:10.1175/JAMC-D-19-0161.1.
2. Wang Y, MacCall B, Hocut C, Zeng X, Fernando HJS. Simulation of stratified flows over a ridge using a lattice Boltzmann model. Environ Fluid Mech. 2018;20:1333–1355. doi:10.1007/s10652-018-9599-3.
3. Wang Y, Huynh G, Williamson C. Integration of Google Maps/Earth with microscale meteorology models and data visualization. Comput and Geosci. 2013;61:23–31.
4. Wang Y, Williamson C, Garvey D, Chang S, Cogan J. Application of a multigrid method to a mass consistent diagnostic wind model. J Appl Meteorol. 2005;44:1078–1089.
5. Huynh G, Wang Y, Rehovicova, K. Mobile app development for the Three-Dimensional Wind Field (3DWF) model on Android and Windows platforms. Army Research Laboratory (US); 2019 Sep. Report No.: ARL-TR-8798.
6. Node.js: an opens source, cross-platform JavaScript runtime environment [accessed 6 Sep 2023]. https://nodejs.org/en.
7. Farr T, Rosen P, Caro E, Crippen R, Duren R, Hensley S, Kobrick M, Paller M, Rodriguez E, Roth L, et al. The shuttle radar topography mission. 2007: Reviews of Geophysics 45, RG2004 [accessed 6 Sep 2023]. doi: 10.1029/2005RG000183.

List of Symbols, Abbreviations, and Acronyms

3-D	three-dimensional
3DWF	Three-Dimensional Wind Field
ABLE-LBM	Atmospheric Boundary Layer Environment-Lattice Boltzmann Model
ARL	Army Research Laboratory
BIL	band interleaved by line
DEVCOM	US Army Combat Capabilities Development Command
EJS	Embedded JavaScript
ENU	East, North, and Up
GIS	Geographical Information System
GUI	graphical user interface
HTML	HyperText Markup Language
IE	Internet Explorer
NPM	Node Package Manager
SRTM	Shuttle Radar Topography Mission
USGS	United States Geological Survey
WGS84	World Geodetic System 84
XML-JS	Extensible Markup Language-JavaScript

1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
1 (PDF)	DEVCOM ARL FCDD RLB CI TECH LIB
4 (PDF)	DEVCOM ARL FCDD RLA N B RIVERA FCDD RLA NA L DAWSON Y WANG E MARK