Prometheus – the Canadian Wildland Fire Growth Model Background and Problem Statement

Rob Bryce
***Prometheus* Software Engineer**
Aldergrove, British Columbia

In April 1999, a proposal was submitted to the Foothills Model Forest (FMF) for the development and application of a wildland fire growth model. The proposal objective was to develop, customize and implement a deterministic fire growth model to allow for operational and strategic assessments of spatial fire behaviour potential in Canadian landscapes. Software engineering of *Prometheus* began in February 2000. The Alberta Forest Protection Branch is now leading this national interagency initiative.

Fire growth modeling involves the simulation of fire spread across landscapes that have spatial variations in fuels and topography. Fire Behaviour Prediction (FBP) fuel types are input into the *Prometheus* model as ASCII grids. Slope, slope azimuth and elevation ASCII grids are also used to integrate terrain effects. *Prometheus* incorporates weather stream data that can vary daily or hourly.

Prometheus is a spatially explicit, deterministic fire growth model. The foundation of the *Prometheus* model is the FBP Sub-System of the Canadian Forest Fire Danger Rating System (CFFDRS), and the wave propagation algorithms developed by Richards (1990, 1999).

In 1678, Christian Huygens, a Dutch astronomer, mathematician and physicist concluded that every point of an advancing light wave becomes the source of new light waves. Huygens' principle of wave propagation was first applied to propagate fire growth in 1975. Since then, several wave propagation fire growth models have been developed (SiroFire in Australia, FARSITE in the United States, and *Prometheus* in Canada). These models use the same approach: the fire front is composed of an ordered list of vertices. Each vertex (or point) on the fire perimeter grows as a firelet (i.e. elliptical wavelet). The vertices move at discrete time steps based on the information at that location, plus the relative locations of its neighbouring points. Collectively, the propagated firelets produce fire perimeters at specific instances in time.

Two difficult problems need to be solved in this application:
1. Each vertex obtains information about fuel, weather, and topography specific for that location. As the vertices propagate through the landscape, the fire front can form complex crossovers, loops, and knots. This is partly because each vertex only considers its immediate neighbours. Fires may also merge with other fires, or within themselves (e.g. burning around a lake). It is necessary to remove these artifacts because they affect the display and execution times of the model, as well as the accuracy of secondary calculations such as total area burned.

The current solution to detect interior vertices implements a “winding number” method (dating to Meister 1769 and Mobius 1865) (Richard, Bryce 1995). Successive algorithms locate and remove loops existing inside the main fire perimeter (areas which have already burned). The current algorithms determining whether vertices are active or inactive are computationally correct but do not address all vertex behaviours exhibited by the wave propogation algorithms, indicating that the original assumptions have been broken. The problem is further compounded because the propagated vertices must also stop when they encounter a vector fuel break (e.g. a river). A new algorithm is proposed to deal with these broken assumptions, but it needs to be formulated as a proof (or replaced with a superior algorithm) before being implemented.

2.) Another important challenge exists in this problem domain. User error can adversely affect the model’s output.

- All vertices advance using the same calculation time step, which is user-specified. Due to different rates of spread, vertices at the head of the fire advance across the landscape quickly; those at the back move correspondingly slower.
- Two thresholding techniques, which are user-tuned, are used to introduce new vertices to the expanding fire front. These algorithms concentrate more vertices around complex areas of the fire front and capture the resolution (cell size) of the FBP fuel type raster layer.
- A smoothing model that uses a simple user-defined weighting equation has been introduced to modify the fire front at each time interval to generate a more visually appealing perimeter. There is no attempt to redistribute/re-space existing vertices.

These four parameters can significantly affect the output of a simulation, and represent a user error in the model. For example, if the time step is too small, or if too many vertices exist on the fire front, then artifacts from the underlying grid landscape data will surface. If the time step is too large, or too few vertices exist on the fire front, then landscape features in the grid will be missed. Algorithms to automatically tune or replace these four parameters (to remove this user error), within the confines of a regular statistical display output, are being sought after with no currently viable solution.