Terrain Aware Backtracking via Forward Ray Tracing

Kurt Rhee
Nevados Engineering
San Francisco, California, United States
kurt@nevadosengineering.com

***Abstract*—Trackers with conventional backtracking strategies can incur significant shading and electrical mismatch losses due to the realities of as-built world terrain, and current best methods for slope awareness only account for single slopes in the cross axis and tracker axis directions. Nevados engineering has developed a new methodology for eliminating terrain shading on horizontal single axis tracking solar sites which can handle arbitrary tracker azimuths as well as multiple slopes in any direction, including installations where the slope changes in both the axis and the cross axis directions.**

Keywords—horizontal single axis tracker, tracker, backtracking, ray tracing, terrain

I. Introduction

Horizontal single axis trackers, which are often used in utility scale photovoltaic power plants, can increase the amount of light captured per module compared to a fixed tilt array and can likewise improve said project's economic return. Two factors that may decrease the overall positive effect of horizontal single axis trackers are the amount of grading required to install a standard single axis tracking system and deficiencies in the tracker rotation strategy used to control how the trackers follow the sun.

Nevados single axis trackers are designed specifically to reduce or eliminate the grading requirement by introducing specialized bearings which can handle post to post net angle changes of up to 15 degrees. Within a single tracker there may be some number of bays, each with it's own distinct normal vector. This design, which improves the grading aspect of project economics, by definition also makes the tracker system more vulnerable to cross axis and intra-tracker axis slopes. This vulnerability is solved for by introducing an iterative forward ray tracing algorithm which tests for module shading.

The most basic tracking algorithm available to system and tracker designers, entitled true-tracking, "minimizes the incidence angle between the panel normal (vector) and incoming beam irradiance from the sun" [1]. "This approach lays modules flat at solar noon and tilts modules at steep angles in early morning and late afternoon to face towards then sun when it is at low elevation. At steep tilt and low sun elevation, inter-(tracker) shading causes cell to cell electrical mismatch in modules with a conventional cell layout, resulting in severe nonlinear power loss" [1].

The next improvement of the tracking algorithm, entitled back-tracking, reduces the module angle during the early morning and late evening hours in order to prevent shade from one tracker falling onto another tracker. This is often called "row to row shading". This algorithm assumes that trackers are entirely within a single plane (flat) and will fail to prevent shading when this condition is not met [1].

The third backtracking strategy, which we will call cross-slope aware back tracking, takes into account the fact that one tracker may lie in a different plane (higher or lower) than the tracker directly to its east or west. The angle between these two planes, called the cross-axis slope, extends infinitely and all trackers that fall within the scope of each algorithm run will inherit the same cross axis slope. This methodology can fail to prevent shading in three ways. One, if the cross axis slope is not constant throughout the whole site and the algorithm fails to take into account the variety of slope changes there may be row to row shading. Two, if the two planes which hold the modules are not parallel with one another, meaning there is a slope in the tracker axis direction, the algorithm will not be able to prevent shade. Three, if slope changes from pile to pile within a tracker, a strategy which only takes into account the tracker as a whole will fail to prevent shading from occurring.

Nevados terrain tolerant trackers have bays which can have both cross axes slopes and slopes in the tracker axis direction, as well as intra-tracker variability in pile top elevations. These facts necessitated the development of a new backtracking algorithm which could account for these additional factors.

II. Methodology

The relevant information required to perform terrain aware backtracking in all dimensions begins with the elevation encoded locations points within a given project. The points should describe a set of rectangular polygons which enclose each contiguous set of modules with a common normal vector (a bay of modules).

From this point, sun angles can be determined and a schedule of true tracking angles can be created for each tracker. The methodology described by Anderson and Mikofski [1] has proven very useful.

Looking at each time step individually one can set up the backtracking problem as a traditional forward ray tracer with no screen object necessary through which to trace rays. In the simplest incantation of a backwards ray tracing program such as those used in the movie and video game industries, rays are traced from a "camera", through the pixels on a screen and into the scene and then finally towards a light source. Because a realistic image is not necessary to determine if a shadow is cast by the direct light from the sun, we can remove the camera and screen objects from our problem and trace the light from the light source (the sun) instead of towards it.

Within the scene, a set of rays can be spawned for each bay in a tracker. Each ray can be defined by a direction corresponding to the corner of the bay and an origin corresponding to the unit vector coming from the direction of the sun.

Because each bay of trackers can be described by a plane with a point and a normal vector pointing outward from the module faces, an intersection test can then be performed for a given ray and possibly shaded bays. If the rays of the sun passing one bay cause shade on another bay, then both trackers will be backtracked by 1 degree until no shading occurs.

Due to the fact that ray tracing is a time consuming process, a number of methods can be implemented so that the problem solves in a reasonable amount of time. These include reducing the amount of intersection tests required by choosing only reasonable objects that could possibly block direct irradiance, and having the ray tracing processes run in parallel.

In addition to the utility a ray tracing backtracking strategy presents for tracker systems that have both cross-axis slopes and variability in intra-tracker axis angle, it can also provide additional energy benefit to a system that is affected by shade objects. At sites that are affected by external shade objects such as those that are bordered on the sides by forest or buildings, the same ray tracing logic can be applied to those structures, and shading can be avoided.

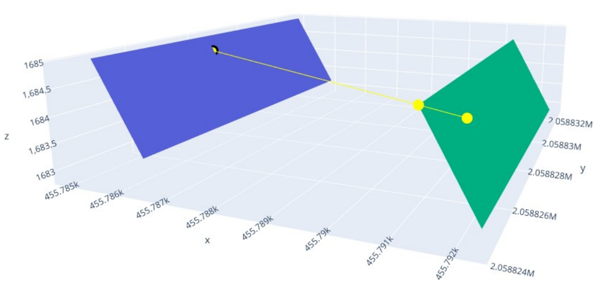

Fig. 1. A ray tracing scene between two bays of modules. In this case the intersection test is positive and backtracking will need to occur.

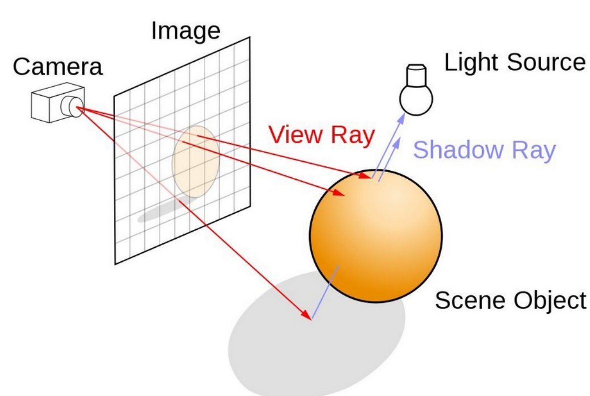

Fig. 2. Simplified backwards ray tracing schematic [3]

III. Conclusion

Forward ray tracing represents a viable way to avoid shading in cases where cross-axis slope is not consistent throughout the site, where intra-tracker axis angle variations occur, where azimuth is not consistent throughout the site, and where object shading or far shading may occur. This method allows Nevados trackers to remain unshaded

Acknowledgment

A word of thanks to Kevin Anderson at NREL for publishing the original methodologies to which this work seeks to add and for helping the author in finding additional reference material for related work.

References

[1] K. Anderson “Maximizing Yield with Improved Single-Axis Backtracking on Cross-Axis Slopes” National Renewable Energy Laboratory NREL/CP-5K00-76023

[2] K.Anderson and M. Mikofski “Slope-Aware Backtracking for Single-Axis Trackers” National Renewable Energy Laboratory NREL/TP-5K00-76626

[3] Henrik “https://commons.wikimedia.org”