Research Article
A GIS-Based Target Tree Management Supporting System Based on Plug-In Technology

[1, 2]Yixiang Wang, [1]Shangbin Bai, [2]Yuanchang Lu and [2]Shougong Zhang
[1]Zhejiang Provincial Key Laboratory of Carbon Cycling in Forest Ecosystems and Carbon
Sequestration, Zhejiang A and F University, Lin'an 311300, Zhejiang, China
[2]Institute of Forest Resource Information Techniques, Chinese Academy of
Forestry, Beijing, 100091, China

Abstract: Traditional forest management models based on stand or landscape level should be transformed to concluding an individual tree level at the same time. Target tree management individual operation method based on close-to-nature sivlculture need to develop to implement such transformation. Aiming to complicated analysis, large computing and the status of lack of effective analysis tool for the operation method, we tried to develop GIS-based plug-in Target Tree Management system Supporting System (TTMSS) to meet this requirement combing plug-in technology of .Net framework, C# and ArcGIS Engine 9.3 components kit. A target tree management individual operation method was established by designing target trees and interference trees decision processes and methods based on Geographic Information System (GIS). The results showed that the operation method improved the feasibility and reduced the difficulty of the target tree management with GIS technology to help improve the operations on the individual tree level. It was proved that TTMSS was an effective tool for target tree management. Under this plug-in mode, the users merely need to do is add or replace plug-ins if one wants to expand or upgrade the system, extremely improve TTMSS's portability and expandability.

Keywords: Geographic Information System (GIS), plug-in technology, target tree management, tree level

INTRODUCTION

Traditional forest management models based on stand or landscape level should be transformed to concluding an individual tree level at the same time (Lu *et al.*, 2009, 2010). Target tree management within the close-to-nature silviculture is a single-tree management system which tries to understand and respect nature, make full use of the growth potential of stand regeneration for ensuring both ecological and economic goals and for producing forest goods as much as possible by minimizing investment in forest management on the basis of maintaining a stable ecosystem (Abetz and Klädtke, 2002; Lu, 2006). The traditional forest management focusing on forest tending is to identify and mark "needless trees", while the target-tree management system is to determine and mark "need trees," i.e., target trees (Lu 2006; Lu *et al.*, 2009). During the whole silvictuture in target tree management all forest actions are regarded as the center of the target trees, including their growth, regeneration, protection and utilization. It is necessary for field staff to find which trees can be regarded as target trees and how to releases (cutting interference trees), but in practice, it is often difficult. One of the mainly reasons are the spatial relationship and its computation among trees cannot be computed by hand.

The target tree management is implemented based on forest spatial structure and spatial relationship between the target tree and the surrounding neighboring trees. The field of forest spatial structure is in the continuous development that new analytical methods will continue to emerge and these methods can be used in the target tree management. At present, using plug-in framework mechanism to develop application software is a better choice as for such professional software. The component pattern of "One modified the overall compilation" cannot adapt all requirements of a GIS-based plug-in Target Tree Management system Supporting System (TTMSS). As for any software, no matter how detailed investigation on users and demand analysis made by project developer before developing are, it cannot meet all requirements of each user. The plug-in technology is a software architecture technology and a higher level of code reuse (Agarwal *et al.*, 2005). Plug-in software development framework is clearer and more flexible from the aspect of software architecture and has the ability of overlay new functionality with the least cost from the aspect of development thinking (Jiang, 2008). If we apply the plug-in technology

Corresponding Author: Yixiang Wang, Zhejiang Provincial Key Laboratory of Carbon Cycling in Forest Ecosystems and Carbon Sequestration, Zhejiang A & F University, Lin'an 311300, Zhejiang, China

mechanism to develop TTMSS, it will have better reusability, easy to be expanded in the future.

This study aims to apply GIS technology to select target trees and interference trees properly to improve target tree management at tree level. Specifically, the objectives are:

- To build a GIS application framework based on plug-in technology
- To design decision process that is used to select target trees
- To design interference trees decision module and develop methods for quantifying crown competition level of a tree and its neighbor trees

DEVELOPMENT OF THE GIS-BASED TARGET TREE MANAGEMENT SUPPORTING SYSTEM

Plug-in technology: A system can be seen as different components by functional composition from the perspective of application program functionality. For an example, the typical forest resources management information system consist map control, query, resource data updating, statistical summaries, mapping, etc., modules, while the different modules corresponding to the different plug-in. A system based on plug-in technology divides the whole application program into host program and plug-in object in the process of software design and development. Host program can call plug-in object and plug-in object can realize its own logic in host program. Their interaction is based on a common communication contract. In this study a GIS-based Target Tree Management System (TTMSS) was built within the .NET platform which support for plug-in technology by reflection technique. The use of reflection technique is a collection of classes in the namespace by System. Reflection, which provides a good object model that can use the Assembly to dynamically load or unload an assembly manifest listed in the module. If we need some new function of TTMSS, we can add or adjust functions only by adding or modifying plug-ins in accordance with the appropriate design of interface specification by avoiding without changing host program in order to achieve a dynamically extensible application. Plug-in itself can be separate tested and deployed without recompile the entire application as component technology as. With this technology, the extension points of the application framework can dynamically load and build by plug-in procedures assembly to achieve the extended TTMSS.

Framework of GIS application in TTMSS: .Net 3.5 Framework and ArcGIS Engine 9.3 were used to develop TTMSS. The former provides the components of general data processing and logical relation for TTMSS while the latter provides the necessary map components and map data manipulation class for TTMSS. With the ArcGIS Engine development tools, we can obtain the functionality provided by ArcGIS to develop GIS application and customize this system. Logically, there are five parts in the components of ArcGIS Engine's component library, that is, Base Services, Data Access, Map Presentation, Developer Component and Extensions. Base Services includes the core component Arc Objects in ArcGIS Engine. Data Access contains all the interfaces and class components of Geo Database for accessing grid data or vector data. Map Presentation contains the components of GIS application program to be used for data display and data symbolic. Developer Component contains all visualized controls needed for rapid development, such as Symbology Control, Globe Control, Map Control, Scene Control, TOC Control, Toolbar Control and the like. Extensions contains many advanced functions such as space analysis, network analysis and data interoperability. The framework of GIS application in TTMSS is to visualize forest geographic data, spatial analysis, data processing and output. Therefore, we use the Geo database, Geometry, Controls, Page Layout and TOC Control to build the framework of GIS application in TTMSS.

Framework of TTMSS: In this study a GIS-based Decision Support System (DSS) was built for carry out target tree individual operation method based on close-to-nature sivlculture by integrating a target trees decision model and an interference trees decision model. The close integration between ArcGIS Engine 9.3, C# development language and Visual Studio .NET 2010 provides many IDE plug-ins to integrate with Visual Studio .NET 2010 closely and enables it easier for developers to edit ArcGIS Engine program based on Visual Studio .NET 2010.

The framework design of TTMSS consist host program, public class library, plug-in engine and plug-in object (Fig. 1). Host program is the entrance of TTMSS and the dependent object of plug-ins, which can be defined as a platform to resolve plug-in object and commission plug-in object events by association to generate a variety of objects such as buttons, tools, toolbars and menu, etc. and also calling some features in public class library. The public library is a set of classes and functions containing some common functions, such as map rendering, which can be called by host program and plug-ins. Plug-in engine parse plug-in program assembly, extract that contains the plug-in type information to generate the corresponding plug-in object which will be stored in the plug-ins collection and transited to the interface program. Plug-in object is to responsible for realize the specific
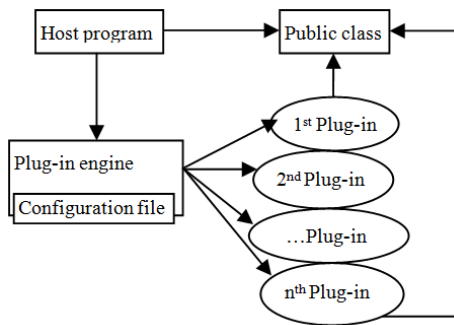
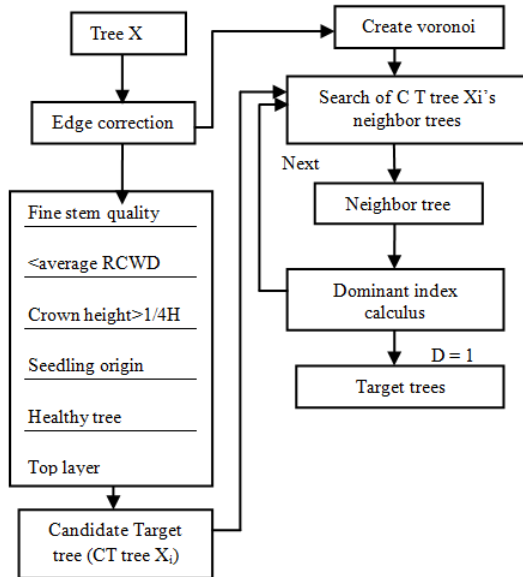Fig. 1: Framework chart of TMSS on plug-in technology

Fig. 2: Flow chart of target trees selection in the forest

functions of the plug-in framework which can be parsed by plug-in engine and called by host program. TTMSS made use of the I Command, I Tool IToolBarDef IMenuDef and IDockablewindowDef defined by the plug-in engine to add the developed plug-ins to the system.

Tree classification in a stand: In order to implement target tree management operation method, we classify each tree in a stand into one of five categories:

- **Target tree:** Dominant tree in the stand, future crop tree. The function is to complete natural seeding and not cut until target diameter reach.
- **Interference tree:** Dominant and co-dominant tree Trees that influence the growth of target trees and need to be cut.

- **Special tree:** Tree that can increase mixed species richness and maintain biodiversity and become wildlife habitat or has the ability of natural seeding.
- **Regeneration:** Next generation trees
- **Ordinary trees:** The remaining stems that support and improve the above trees provide shade or protect other trees. Whether it is cut or not needs to consider managers' timber demand.

This key point of operation method is expected to address the following two questions:

- How to select the target trees
- How to select interference trees

Implementation of the target trees decision model: We select target trees from stem quality, Ratio of Crown Width to Diameter (RCWD), crown length, origin, healthy status and position in the story and dominance intensity. Figure 2 shows the requirements for target trees.

If the tree X_i in the forest is fine stem quality, lower than RCWD, seedling origin, healthy, in top layer and its crown height larger than 1/4H, it is the candidate for target tree (CTtree X_i); otherwise it cannot be selected as a target tree. To calculate its dominance intensity, we create Voronoi to find CTress's neighboring trees. Each CTtree's neighbors need to be identified to apply the dominant index model. The variables needed in this algorithm include diameter and crown width or height. If dominant index = 0 (D = 1), CTtree X_i is selected as a target tree:

- **Search each tree's neighborhood with the Voronoi diagram:** Assuming *X* trees in a stand, which will be the X_i's neighbor trees? If the tree in a stand is regarded as a point, the problem becomes: given a set *X* of points in a metric space M and a query point $X_i \in$ M, find the neighbor point from *X* to X_i. Various solutions to this problem have been proposed (Bella, 1971; Hegyi, 1974; Pukkala, 1989; Pretzsch, 1992). One method is to analyze the space partitioning among individual points using Dirichlet-Thiessen tessellations and then to divide polygons surrounding the individual points (Georgy, 1907; Upton and Fingleton, 1985). We create Voronoi tessellation (Fig. 3) to give a detailed description of the position and shape of individual plants in relation to the number and proximity of their contiguous neighbors in TTMSS. Figure 3 shows the subject tree's neighborhoods are the 1st, 2nd, 3th, 4th, 5th, 6th trees within the Voronoi polygon method.
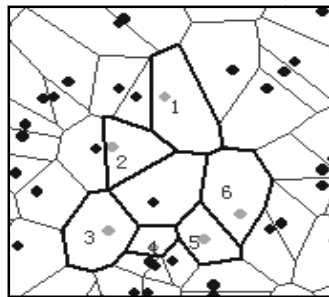
Fig. 3: Subject tree and neighboring trees

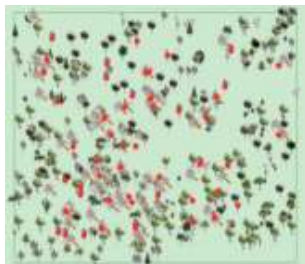Fig. 4: Individual trees spatial distribution map in a stand

Fig. 5: Target trees and interference trees spatial distribution map in a stand

- **Calculate dominant index of an individual tree:** The dominant intensity algorithm of an individual tree can be defined as the proportion of the number smaller than subject tree to its total number of the neighboring and can be expressed as Eq. (1):

$$D_i = \frac{1}{n} \sum_{j=1}^{n} k_{ij} \qquad (1)$$

  K_{ij} = {0, if the ith tree is smaller than its jth neighboring tree, 1 otherwise

Comparison option item is height, diameter or crown area. It reflects the tree's dominance in its spatial location.

Take Fig. 3 as an example, there are 6 neighbor trees for the 0th tree and then n is 6. If the diameter of the 0th tree is larger than any one of the 6 trees, its D value is

the maximum value as 1. The larger D value means the lesser neighbor trees larger than the centered tree and the greater of the dominance.

Figure 4 is the capture image of TTMSS, which means the spatial distribution map of the individual trees in a stand divided by Voronoi diagram. As a result of target trees decision model application for a 40×40 m plot, red points represent the target trees (showed in Fig. 5).

Implementation of the interference trees decision model: Once a region of competition and neighboring trees is defined, the second issue that must be addressed is which neighboring trees influence target tree's growth and intensity. After the target trees were determined, we need to calculate the relationship between each target tree and its neighbors. This kind of interaction between trees can be quantified by using Horizontal Competition Index (HCI) and Vertical Competition Index (VCI) and they are calculated in TMSS for each target tree/neighbor pair so that interference grade can be obtained. If the calculated interference grade meets the forest manager's requirements, the neighbor trees are classified as interference trees that need to be cut:

- **Calculate Horizontal Competition Index (HCI):** Horizontal competition index is the ratio of the crown overlap (or gap) width between subject tree and neighboring tree to the crown width of subject tree. HCI is used to describe competition between target tree and its interference tree in the horizontal direction:

$$HCI = (CW_s + CW_i - D_{si}) / CW_s \qquad (2)$$

  where,
  CW_s : The crown width of the subject tree on the direction to the ith neighboring tree
  CW_i : The crown width of the ith neighboring tree on the direction to the subject tree
  D_{si} : The horizontal distance between subject tree and the ith neighboring tree

  When crowns almost touch each other, HCI is close to zero; when there is a gap, the HCI becomes negative; for overlapping crowns, HCI becomes positive. A larger HCI value indicates stronger competitive. If the tree is released by thinning to the subject tree on one side, the HCI becomes smaller.

- **Calculate Vertical Competition Index (VCI):** Vertical competition index is the ratio of the height and location difference between subject tree and i^{th} neighboring tree to the crown length of subject

Table 1: A summary of the levels of interference competition

	1/2<HCI	1/3≦HCI≦1/2	0≦HCI<1/3
VCI<0	1	2	3
0≦VCI≦0.5	4	5	6
0.5≦VCI≦1	7	8	9

tree. VCI is used to describe competition between target tree and its interference tree in the vertical direction:

$$VCI = (H_s - H_i + \Delta h_{si}) /CL_s \quad (3)$$

where,

H_s : The height value of subject tree
H_i : The height value of the ith neighboring tree
CL_s : The crown length of subject tree
Δh_{si} : The location altitude difference between them

If $\Delta h_{si}>0$, it shows that the location altitude of subject tree is higher than that of the ith neighboring tree. Smaller VCI indicates more intense competition, while larger VCI means less intense competition.

- **Determine the level of interference competition:** Combination of HCI and VCI determines the level of interference competition between target trees and neighboring tree (Table 1). Meanwhile, in the course of determining a competition level we assume that VCI be superior to HCI.

Table 1 indicates that the interference trees are selected by the level of competition intensity in the horizontal and vertical directions between target tree and neighboring trees. Forest managers can determine which interference level is needed to select interference trees according to his stand operation requirements. The competition level of the neighboring trees is divided into nine grades, of which the first level of interference means the competition with the target tree is the most intense, the second level represents the level of competition is less intense and so on and the ninth level represents the minimum competition.

In the Fig. 4, as a result of interference trees decision model application for a 40×40 m plot, axe symbol represent the interference trees.

DISUCUSSION AND CONCLUSION

We developed GIS-based Target Tree Management System (TTMSS) by applying .Net platform plug-in technology, C# development language, combined with the ArcGIS Engine 9.3 component development kit to achieve a "platform+plug-in framework mode. Users only need to add or replace plug-in to complete the expansion and upgrade of the system by avoiding changing host program, greatly improve the portability and expandability of TTMSS.

Forest management has evolved from a relatively classical timber production approach to multipurpose management for reconciling various conflicting demands between timber and non-timber resources (Emin *et al.*, 2008). Traditional forest management models were applied to the stand level or landscape level based on average variables of the stand such as average DBH, average height, basal area, mean growth, which were utilized to assign stand operation by forest managers (Moore and Lockwood, 1990; Yoshimoto *et al.*, 1994; Lu *et al.*, 2009, 2010). The manual selection target trees and interference trees method is time-consuming and labor intensive, requiring experts to observe trees from different angles by comparing the difference among trees. The selection results vary, depending on different times and different experts. In this study, TTMSS tries to help forest managers identify target trees to be protected, tended and interference trees to be cut based on the size inequality and species difference among individual trees in the stand. TTMSS allows the manager to consider all trees simultaneously and the optimal solution found has the highest probability of being the best.

The individual operation method based on the target trees and interference trees decision model might be useful for standardizing silvicultural prescriptions among forest managers. It could be used as a tool for decision-making processes in forest management and applied in a general manner. The main shortcoming of TTMSS is that too much tree variables and a large amount of data are needed. When the study area is large, time and costs of gathering data will be a constraint. Future studies will extend target tree management as a promising approach to meet the criteria for sustainable forest management.

REFERENCES

Abetz, P. and J. Klädtke, 2002. The target tree management system. Forstw. Cbl, 121:73-82.
Agarwal, R., A. Sasturkar, L. Wang and S. Stoller, 2005. Optimized run-time race detection and atomicity checking using partial discovered types. USA: 20th IEEE/ACM International Conference On ASE, pp: 233-242.

Bella, I.E., 1971. A new compe tition model for individual trees. For. Sci., 17(3): 364-372.

Emin, Z.B., K. Sedat and A.Y. Haci, 2008. Comparing multipurpose forest management with timber management, incorporating timber, carbon and oxygen values: A case study. Scandinavian J. For. Res., 23: 105-120.

Georgy, V., 1907. Nouvelles applications des paramètres continus à la théorie des formes quadratiques. J. für die Reine und Angewandte Mathematik, 133: 97-178.

Hegyi, F., 1974. A Simulation Model for Managing Jack-Pine Stands. In: Fries, J. (Ed.), Growth Models for Tree and Stand Simulation. Royal College of Forest, Stockholm, Sweden, pp: 74-90.

Jiang, B.T., 2008. Plug-in GIS Application Framework Design and Implementation based on C# and ArcGIS Engine 9.2. Publishing House of Electronics Industry, Beijing.

Lu, Y.C., 2006. Theories and Practices on Close-to-Nature Silviculture. Science Publishing House, Beijing.

Lu, Y.C., S.G. Zhang, X.D. Lei, J.K. Ning and Y.X. Wang, 2009. Theoretical basis and implementation techniques on close-to-nature transformation of plantations. World Forestry Res., 22(1): 20-27.

Lu, Y.C., S.Q. Luan, S.G. Zhang, H. Bernhard, X.D. Lei, Y. Bao, 2010. From normal forest to close-to-nature forest: multi-functional forestry and its practice at national, regional and forest management unit levels in Germany. World Forestry Research, 23(1): 1-11.

Moore, T. and C.G. Lockwood, 1990. The HSG wood supply model: Description and user's manual. Report PI-X-98. Petawawa Natural Forestry Institute.

Pretzsch, H., 1992. Konzeption und Konstruktion von Wuchsmodellen fär Rein- und Mischbestände. Forstl Forschungsber München, 115: 358.

Pukkala, T., 1989. Methods to describe the competition process in a tree stand. Scandinavian Journal of Forest Research, 4: 187-202.

Upton, G.J.G., B. Fingleton, 1985. Spatial data analysis by example. Point Pattern and Quantitative Data. Wiley, Chichester, pp: 410.

Yoshimoto, A., J.D. Brodie and J. Sessions, 1994. A new heuristic to solve spatially constrained long-term harvest scheduling problems. For. Sci. 40: 365-396.