Development of a new computer program for designing forest roads

Nevečerel, H., Lepoglavec, K., Papa, I. Pičman, D., Pentek, T.
UNIVERSITY OF ZAGREB - FACULTY OF FORESTRY
Svetosimunska 25, Zagreb, Croatia
hnevecerel@sumfak.hr

Abstract:
First use of a computer program for the process of designing forest roads in Croatia was in the year 1988. It was a computer program named SILVIA (Silvae VIA) written in Quick Basic programming language and developed by forestry experts from the FA Delnice. Since 1999 for computer designing of forest roads in Croatia, software named "CESTA" (developed by a Slovenian company SoftData) has been in use. However, this program isn't the best long term solution for designers - foresters in our country. Today in the world there are many programs that are used for forest road design, but the vast majority of them were developed primarily for the design of public, rather than forest roads. The new program is written in programming language C ++. User Interface is made at WxWidgets Opensource platform. A computer program consists of 5 units - subprograms: Main, Calc, Util, Database and Printout. Current version is composed of 7 different views that show: Horizontal Alignment, Vertical Alignment, Cross Section, Main Vertices, Cross Lines, Longitudinal Vertices, and Sketch. With the module for forest road design that is now active, possibility of new development is still open. This applies to, for forestry necessary and useful, modules such as: module for the planning of forest roads, module for maintenance and repair of forest roads, etc.

Key words: forest roads, designing, computer programs for forest road designing, Republic Croatia.

1 Introduction and issues

Forest roads are essential infrastructure and one of the fundamental preconditions for the contemporary management of forest ecosystems. The application of present day techniques and technology (Nevečerel, 2010) is unthinkable and economically unfeasible for the exploitation of forest resources without the existence of forest roads. For the proper and rational management of forest ecosystems (Pentek, 2010), the presence of an optimally positioned network of primary and secondary forest road infrastructure is necessary. According to Ryan et al. (2004), the establishment of an optimal network of forest roads in the field takes place through the following phases:

- planning,
- project designing,
- route laying (field work),
- construction,
- deriving of underground and aboveground drainage and maintenance structures.

The planning, design and construction, with supervision and maintenance are imperative and mutually connected and inseparable phases of forest roads network establishing in the field. In addition to these essential phases, we two occasionally included phases are distinguished: reconstruction, which is carried out to improve the standard of existing forest roads, and the roads removal, at the moment when the forest road proves to no longer be needed.The design of forest roads, in the above mentioned essential elements, represents an unavoidable phase in the process of developing new primary forest roads. Designing a certain forest road means its conception, description and graphical and calculation overviews. Only a fully developed main forest road project can be analysed and, once accepted, construction can begin. The design of forest roads consists of collecting general and technical data, determining the route and developing the project. Field measurements and data collection during road design are carried out using classical methods – theodolites and grade levelling and the use of contemporary methods with geodetic measuring stations. The use of various GPS receivers is still in the experimental phase. Once field measurements are completed, the final product, i.e. the main forest road project, is developed by computer.

2 Study objectives and methods

2.1 Study objectives

The main objectives of this paper are divided into the following segments:

- analysis of existing computer programs used for the design of forest roads,
- development of a theoretical model of a computer program for the forest road design,
- development of a computer program for the design of forest roads.

2.2 Study methods

Analysis of existing computer programs for designing forest roads
Existing computer programs for the design of forest roads will be obtained, tested in real situations and their individual operations and functionalities analysed. Due to the specificity of the issues of designing forest roads, the program will be critically analysed through the prism of the needs of the forestry industry. The various approaches to program development, from the perspective of use of the final templates, will be one of the most significant elements of the analysis. The intended use of each program will also be examined. Designing forest roads differs substantially from the design of public roads, and thus special attention will be directed at defining the basic (primary) intentions of each program.

Development of a theoretical model for the computer program
The functionality of the final computer program will significantly depend on the theoretical approach to the issue of forest road design. Developing the structure of the program will determine the operability of individual elements and will ultimately unite all the useful functions into a single balanced whole – a computer program for the design of forest roads. The proper defining of the operative entity and division of tasks within will make the program faster and easier to use for the user. The theoretical model includes all known algorithms for the forest road design. The proper selection of algorithms will increase the precision of the computer program and make it a useful tool in resolving all known situations.

Selection of programming language and code for the new computer program
The development of contemporary programming language provides many opportunities to create new computer programs. The proper choice implies having knowledge of the existing solutions available, in which the future appearance of individual algorithms will determine the functionality of the final program. The C++ programming language was selected as an important and virtually irreplaceable tool with a broad spectrum of capacities, thereby allowing programmers to find a solution to every problem when developing their ideas into programs.

There are four important properties of the C++ programming language that make it objectively oriented:

- encapsulation,
- data hiding,
- inheritance, and
- polymorphism.

All these properties contribute to the creation of an objectively oriented programming paradigm. The basic idea is to divide the program into a series of separate units which then "cooperate" to resolve the problem. Instead of specialized procedures that carry out the operation with data, instead we work with these units, called classes, which unite the operations and data. In this approach, what is important is what the classes do, and not how they do it. Thereby, it is made possible for one class, if needed, to be removed or replaced with another, better class, if they both do the same thing.

The key to achieving this objective of joining data and operations is called encapsulation. In this, the data are "private", i.e. specific to each class, and may not be accessible to other parts of the program. This property is called data hiding. Each class exclusively offers its environment the data that are necessary for the class to be used. These data, together with the operations that are either accepted or rejected make up the interface of the class. As such, the programmer who will use the given class no longer has to think about the way the class functions, he just simply needs to "ask" the class for a certain service.

While solving a problem, we can start from the beginning (conditionally), or we can use an existing component that is near to the solution, and just add new possibilities. This is called reusability and is a very important property of the selected C++ programming language. New programming components are said to have inherited the properties of the components from which they arose.

The user needs not worry about which version of the component he is using. He will only request the service from the component, and it will execute the given function in the appropriate manner. This property is called polymorphism.

Therefore, what makes C++ a very suitable programming language in the general intent for developing complex programs is the possibility of simply introducing new types, and the subsequent addition of new operations.

3 Study area

Field measurements (data collection) were conducted in the state forests of the Republic of Croatia under the management of the company Croatian Forest Ltd. (Hrvatske šume d.o.o. Zagreb). The Gospić Forest Administration was selected as the representative forest district administration. It is composed of forests with varying stand and habitat conditions, which guarantee the diversity of the collected data and final product of the computer program for the design of forest road construction projects. The area of the Gospić Forest Administration is divided into 11 Forest Offices: Brinje, Otočac, Korenica, Perušić, Donji Lapac, Gospić, Sveti Rok, Udbina, Gračac, Karlobag and Vrhovine.

Field measurements were conducted within four Forest Offices (Brinje, Otočac, Korenica and Udbina). A total of 63 forest road routes were measured. Forest Office Brinje is divided into 7 Management Units, and a total of 18 main forest road projects were measured and developed. Forest Office Otočac consists of 12 Management Units, in which 1 main forest road project was developed. Forest Office Korenica consists of 10 Management Units, and 16 forest roads were designed in this unit. Forest Office Udbina consists of 6 Management Units, in which 28 main forest road projects were measured and developed.

4 Results and discussion

4.1 Development of a theoretical model for the computer program for designing forest roads – fundamental assumptions

The development of a new application should be approached very systematically, as the development of a high quality "frame" is the foundation of functionality of the future computer program for forest road design. In line with this, many foreign computer programs dealing with the issues of forest road design were analysed, including those developed for the design of public roads that are also used in the design of forest roads. In order to facilitate the development of the program and to reveal and eliminate any possible errors, the development of the program was divided into collecting field data and the actual development of the computer program.

The collected field data served to test the functionality of individual algorithms and the program as a whole – testing the accuracy and applicability in various operating systems. Data were collected in different forest offices and management units to test the program using specific examples in different stand and habitat conditions.

The total length of the selected route of the future forest roads was 87.45 kilometres. Data were collected on approximately 9100 profiles (x, y, and z coordinates) and cross-sections.

Generally speaking, the computer program will consist of the following phases of work:

1. creating of a new project,
2. data entry,
3. situational sketches,
4. design of the longitudinal profile,
5. design of the cross sections.

Creation of a new project
The uniting of all these necessary data is tied to a specific forest road. For each individual project, the technical conditions are determined, such as the number of road lanes, width of pavement and berm, minimum radius of horizontal curves, expansion of the road in curves, maximum permitted longitudinal slope of the level grades, minimum radius of concave and convex vertical curves and minimum interval between vertical curves of the same or opposing directions, etc.

When creating a new project, the use is given the option of selecting between several categories of forest roads with the accompanying technical conditions. Each individual category of forest roads is included in a database, which is easily updated in the case of changes to the existing technical requirements.

Each new project is a separate entity (group of data) that are used in the documentation and completion of the projects and which represent the operative base (archive) that are also accessible to other users of the computer program. Such a database could, in practice, serve for the constant amending and updating of the cadastre of the primary forest road infrastructure. All projects would be archived in one place, and they would be subjected to verification and accessible for various research.

Data entry
This allows that data for existing forest road projects to be entered, manually or automatically (in digital form).

Due to the complexity and diversity of today's geodetic instruments for the collection of field data, it is necessary to simply take the same data and translate them into a clear and readable format. The needs of the final user of the computer program are guidelines for the selection of new supported formats that will simplify the transfer of collected data, and reduce the time needed for their entry.

The majority of the present day program packages are suitable for saving data in *.dxf format or as an ASCII file, which allows for the simple transfer from the most commonly used geodetic measurement instruments to a personal computer.

The results of the field measurements are then entered into the existing or newly created project, and saved in the operative database. It is necessary to verify the quality of the entered data, and to inform the user of the state and quality of the file, so that, if needed, the user is able to immediately react in terms of their correction or supplementation.

The processed data can be exported and translated in *.dxf and ASCII files for the purpose of their use in other computer programs (e.g. for the drafting of spatial analyses).

Situational sketches
The creation of situational sketches – the horizontal alignments, defines the position of the polygonal points of the axial polygon in which the radius of horizontal curves are determined, and the tangents and bisecting lines are calculated. When the horizontal curves are calculated, all elements of the forest road route are drawn into the horizontal alignment.

With the processing of polygonal points, the total length of the forest road route and the stationing of each individual profile are calculated. Once drafted, the horizontal alignment offers the possibility of subsequent amendment.

Data are entered in *.pdf format to facilitate printing on computers that do not have the forest road design program installed. This option is handy when it is necessary to create multiple copies of the project quickly, and to complete individual project tasks within deadline.

Designing the longitudinal profile
Designing the longitudinal profile includes the developing the route height with all field data. Each profile of the forest road route has its code, altitude and appearance in the cross section.

In the existing terrain, the unrounded level grades are added and rounded while abiding by the minimum radius of concave and convex curves, and the interval between curves of the same or opposite direction for the given category of forest road. Determining the vertical curves of the included level grades is provided numerically and graphically.

While designing the longitudinal profile, and with regard to the selection of individual categories of forest roads, it is necessary to warn the users of any possible deviations from the technical conditions.

Designing the transverse profile
The accepted rounded level grade embedded in the longitudinal terrain profile, together with the lateral section of the forest road terrain: the elemental route in the horizontal alignment defines every individual transverse profile that can be altered as the designer sees fit. When designing the transverse profile, it is necessary to show the appearance of the terrain in the direction of movement, and every change to the level grade is made in the transverse sketch. This is where the slopes of cuts and fills are determined, drainage channels are added in and created, the thickness of the pavement construction added with all its parameters, the construction category of materials entered for the selected segments, etc.

The drawing of the transverse profile allows us to calculate the various surfaces of the individual profile. With the help of the surfaces and intervals between profiles, the quantities (of cuts and fills, etc.) can be calculated. Ultimately, the diagram of haul mass distribution is created based on these quantity data.

4.2 Development of an application for designing forest roads

In the development phase, the computer program is divided into five subprograms which separately execute the tasks necessary for the functioning of the overall program. One of the subprograms (*Main*) unites all the information and manages it, while the remaining subprograms have an auxiliary task in order to ensure that the entire program can quickly execute the requested observation.

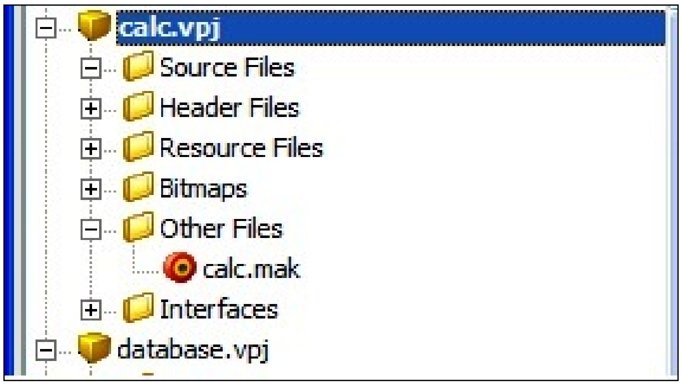


Figure 1: Subprogram Calc in the development phase

The first subprogram (*Calc*) contains all the calculations and represents the mathematical and logical part of the application. The results from every calculation can be shown from each of the user parts of the application. This subprogram represents the fundamental part of the computer program in the sense of application of various calculation methods, and will be created with an open add-on option or replacement in the case of selection of a newer, more precise and better way, and for the purpose of reducing the burden on processor requirements.

In the second subprogram (*Database*), the functions necessary for connection with the database will be implemented. The subprogram will be able to accept a new Module at any time, if the need should arise. The manner of creating the subprogram is in line with the present day principles of designing databases and represents that part of the program that will serve as a database of developed structures, and will be able to accept all projects created with this program, which opens up additional options in the sense of possible user needs.

The third subprogram (*Main*) is the management component, and consists of a graphical user interface and implemented main functions. Functions from the remaining subprograms are called up from this subprogram, and it forms the central subprogram responsible for the execution of all user requests. Development of the entire computer program will lean towards the possibility of accepting external, independent modules, created in line with the user desires and the possibility of implementing new and more demanding tools that will increase the scope of future users.

Furthermore, one of the subprograms (*Util*) was developed with the goal of unburdening the management subprogram, in order to accelerate of the coding – compilation – connection cycle.

Last, but not least important is the subprogram that allows for the printing of all the components of the main forest road project. It is very important to ensure printing of all graphical elements in the selected scale, and the numerical elements on the required forms. Both facilitate and accelerate the development of forest road projects for users.

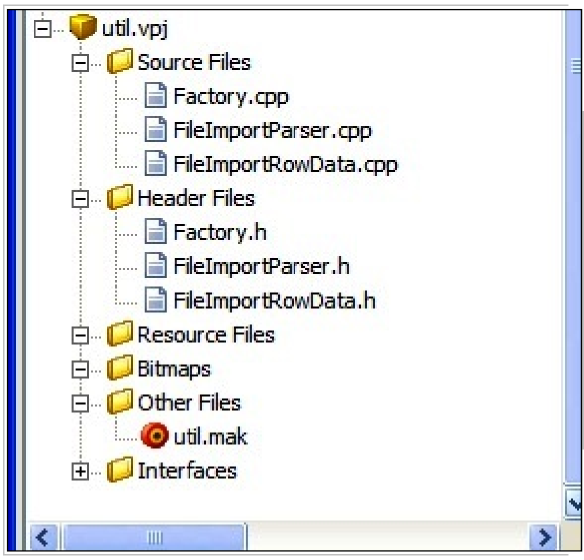


Figure 2: Subprogram Util in the development phase

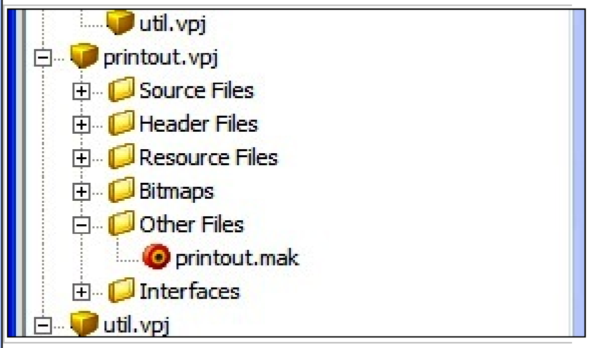


Figure 3: Subprogram Printout in the development phase

The program, printed in C++ programming language, is based on a WxWidgets platform that enables the creation of an application for the following operating systems: Windows, OS X, Linux and UNIX, and supports 32-bit and 64-bit architecture. Applications developed on the WxWidgets platform can also be used on mobile platforms, including Windows Mobile, iPhone SDK and embedded GTK+.

WxWidgets is an opensource platform selected for its:

- openness (it is easily acceptable for users),
- standardization (stability – larger number of projects),
- price – free (the platform was created as an academic project).

Creating the computer program is based on the initial development of the hierarchy where certain classes are that will be defined later and will be assigned specific functions.

4.3 Structure of the computer program

The statistical overview of the computer program with all the obtained elements looks like this:

- Working title of project: htrj.vpw,
- Total number of subprograms: 5,
- Total number of files: 145,
- Total number of lines of code: 22664.

The statistical overview of the computer program by subprograms looks like this:

- Name of subprogram: calc.vpj – 15 files, 1704 lines of code,
- Name of subprogram: database.vpj – 1 files, 81 lines of code,
- Name of subprogram: main.vpj – 115 files, 20246 lines of code,
- Name of subprogram: printout.vpj – 4 files, 139 lines of code,
- Name of subprogram: util.vpj – 10 files, 494 lines of code.

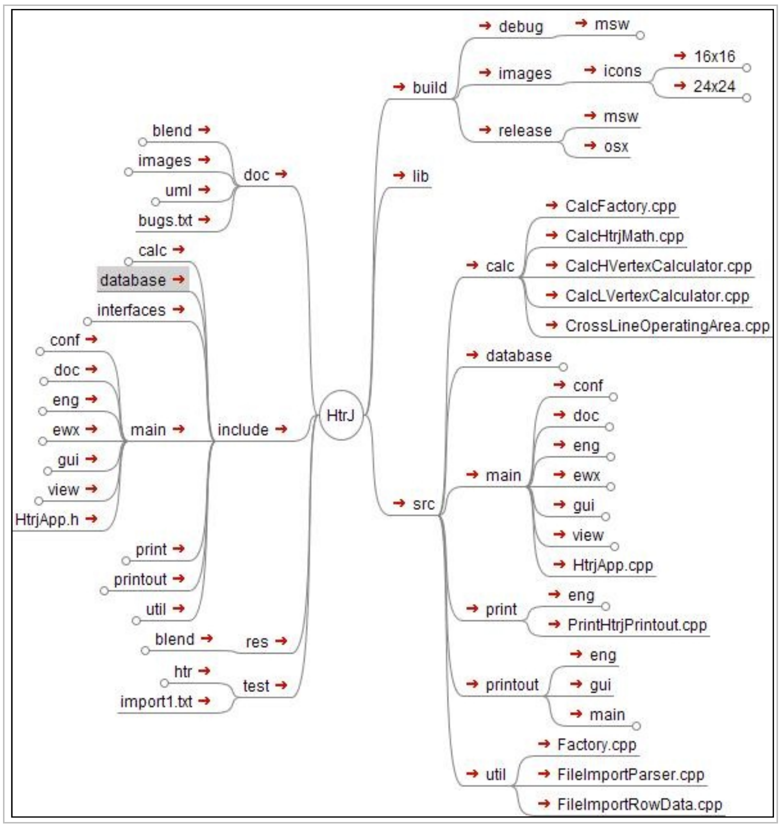


Figure 5: Final structure of the developed computer program with all interconnections

4.4 Final appearance of the developed computer program

The user interface of the new computer program is determined by the user platform WxWidgets, which has predefined settings in a windows setting and the interaction (between the user and application is user friendly). One of the basic problems that needs to be resolved is the speed of the access of each window due to the shortened time of seeking the window, i.e. unnecessary clicks with the mouse. Certain shortcuts are represented by icon, or are found in individual groups of tools. The basic options are taken from the platform options (Wxwidgets), while the appearance of other tools will be created according to user needs. There are seven basic windows, each with its own function, and they are mutually connected by *live-links*, meaning that all data changed in one of the windows is almost instantaneously (calculation time is very short) becomes visible in all other windows.

The first window shows the layout of the forest road – horizontal alignment. This window contains all the functions associated with the horizontal layout of the route, the axis polygons (polygon segments), polygon points (number and codes), width of road surface and banks, profiles (number, code and stationing), road structures (bypasses, turnaround areas, etc.). Graphical modification of the existing data is made possible, so that the user can work further on sections that he is not satisfied with, i.e. to supplement specific atypical situations that could arise in the project development process.

In the horizontal alignment, scrolling with the mouse allows for slow zooming in and out to view details. A panning option is also available to more easily view the desired detail, and in this way, avoids showing details by way of a function.

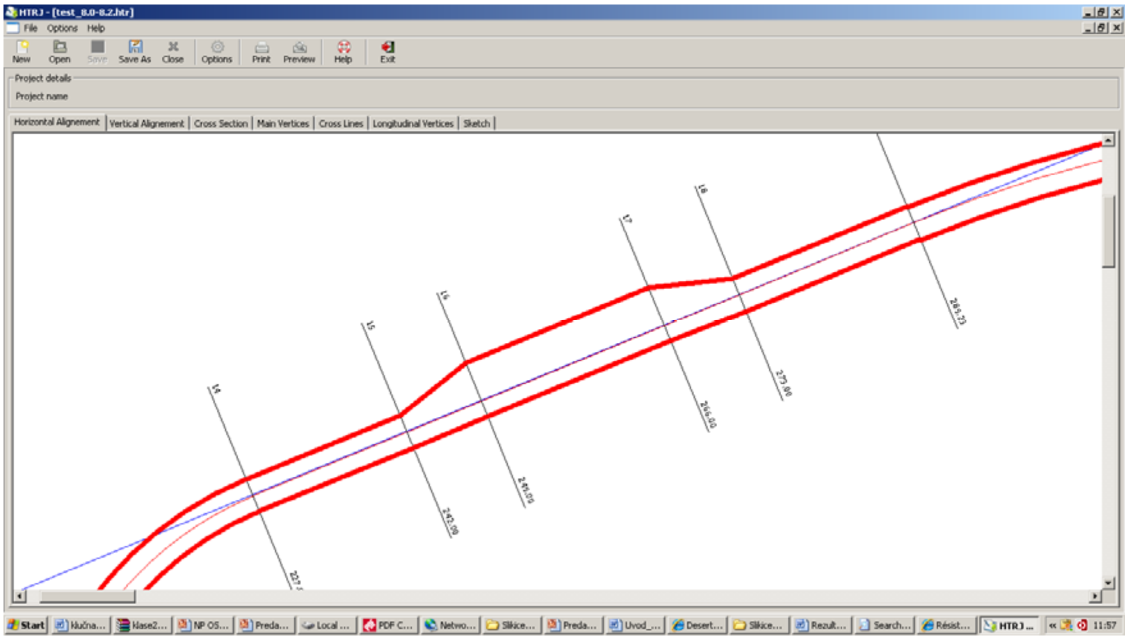


Figure 6: Layout – detail of a full bypass area on the left

Unrounded levels in the longitudinal section of the terrain can be embedded either numerically (with the entry of the position of break points of the level) and graphically (directly using the mouse the position of break points of the level grades are selected). Once the unrounded level grades are accepted, these can be corrected and altered multiple times. With the entry of the radius of the vertical curves at each peak of the unrounded level grade, the vertical curve is drawn in, thereby turning the unrounded level grade into a rounded level grade. Subsequent corrections to the designed rounded level grades are possible, both numerically (using various entry parameters) or graphically (directly).

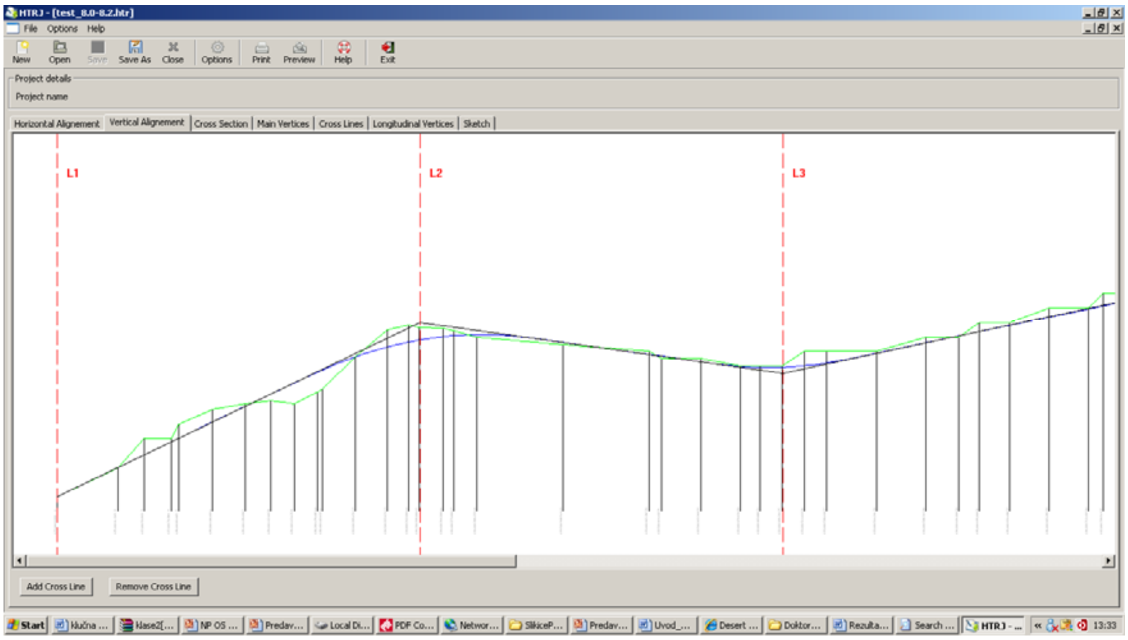
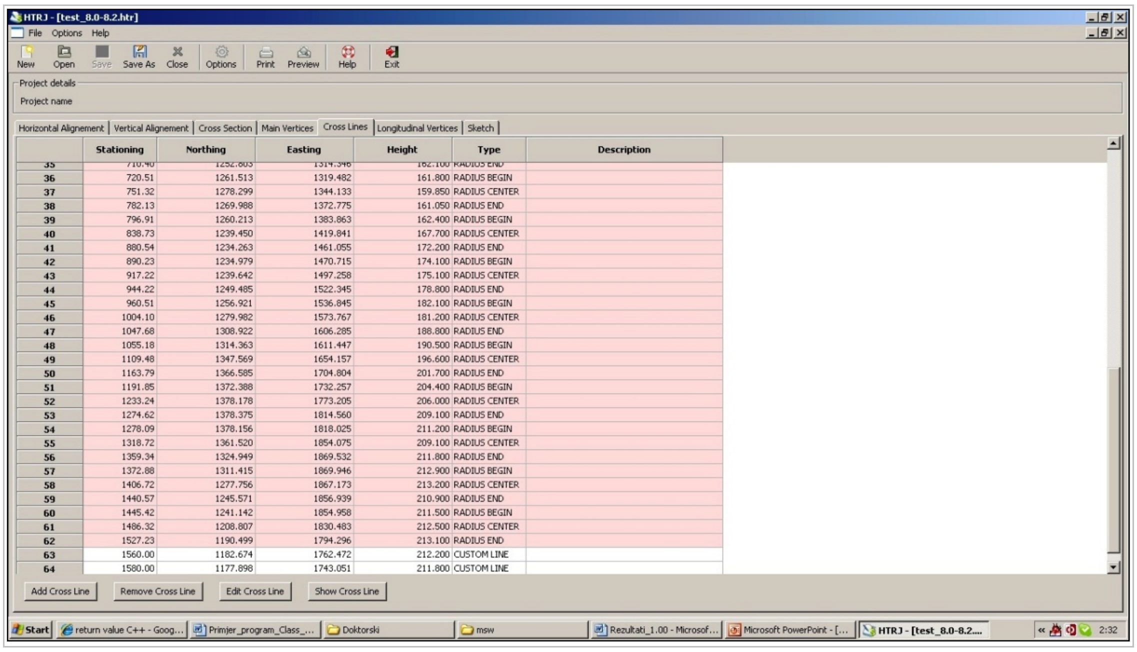


Figure 7: Vertical alignment – transformation of unrounded level grades into rounded level grades

Cross section is the view that shows the actual appearance of an individual profile of the forest road perpendicular to the road axis, and depending on the cross-section of the terrain. Cross sections are drawn (and calculated) based on the normal cross section (in which the basic technical properties of the forest road are defined that are connected to the view of the Cross section). The following are entered and confirmed in this window: slope of cuts, slope of fills, thickness of pavement construction, etc. The Cross section window lists the number and code of the profile, stationing, terrain elevation, level grade elevation, transverse slope of the pavement, and area of each of the components of the transverse profile. The area of the cuts and fills and the value of quantities belonging to individual cross sections are added so as to immediately obtain insight into the distribution of the haul mass of individual segments of the forest road route.

The axis polygon is drawn into the existing data automatically, currently they are defined in the computer program by reading *.dxf and ASCII format, or the entire entry of the polygon points can be conducted manually.

The road route profile is calculated from the polygon and is represented by its spatial coordinates. Each individual profile can be shaped by changing each of the three axes (x, y, z). Each correction to individual data is independent of the other data, as the computer program recalculates the horizontal appearance of the route with every change in this layer of data. The described indicates the possible burdening of the computer if they are entered, the great processor speeds compensate for the frequent calculations, and so there is no influence of the speed of executing the operation. With this, the danger of transferring errors that could arise with manual entry is avoided.



	Stationing	Northing	Easting	Height	Type	Description
35	710.40	1252.803	1314.346	162.100	RADIUS END	
36	720.51	1261.513	1319.482	161.800	RADIUS BEGIN	
37	751.32	1278.299	1344.133	159.850	RADIUS CENTER	
38	782.13	1269.988	1372.775	161.050	RADIUS END	
39	796.91	1260.213	1383.863	162.400	RADIUS BEGIN	
40	838.73	1239.450	1419.841	167.700	RADIUS CENTER	
41	880.54	1234.263	1461.055	172.200	RADIUS END	
42	890.23	1234.979	1470.715	174.100	RADIUS BEGIN	
43	917.22	1239.642	1497.258	175.100	RADIUS CENTER	
44	944.22	1249.485	1522.345	178.800	RADIUS END	
45	960.51	1256.921	1536.845	182.100	RADIUS BEGIN	
46	1004.10	1279.982	1573.767	181.200	RADIUS CENTER	
47	1047.68	1308.922	1606.285	188.800	RADIUS END	
48	1055.18	1314.363	1611.447	190.500	RADIUS BEGIN	
49	1109.48	1347.569	1654.157	196.600	RADIUS CENTER	
50	1163.79	1366.585	1704.804	201.700	RADIUS END	
51	1191.85	1372.388	1732.257	204.400	RADIUS BEGIN	
52	1233.24	1378.178	1773.205	206.000	RADIUS CENTER	
53	1274.62	1378.375	1814.560	209.100	RADIUS END	
54	1278.09	1378.156	1818.025	211.200	RADIUS BEGIN	
55	1318.72	1361.520	1854.075	209.100	RADIUS CENTER	
56	1359.34	1324.949	1869.532	211.800	RADIUS END	
57	1372.88	1311.415	1869.946	212.900	RADIUS BEGIN	
58	1406.72	1277.756	1867.173	213.200	RADIUS CENTER	
59	1440.57	1245.571	1856.939	210.900	RADIUS END	
60	1445.42	1241.142	1854.958	211.500	RADIUS BEGIN	
61	1486.32	1208.807	1830.483	212.500	RADIUS CENTER	
62	1527.23	1190.499	1794.296	213.100	RADIUS END	
63	1560.00	1182.674	1762.472	212.200	CUSTOM LINE	
64	1580.00	1177.898	1743.051	211.800	CUSTOM LINE	

Figure 8: Tabular overview of basic data of forest road route

Once the route of the forest road is defined horizontally, the appearance of the longitudinal profile is the next step, with the setting of unrounded level grades. The shaping and entry of vertical vertex is conducted either graphically or numerically (for greater precision). In the vertical vertex, the radius of the vertical curves is entered (numerically); the objective is to best integrate the rounded level grades of the forest road into the longitudinal profile of the terrain, pursuant to the stipulated technical conditions of those categories of forest roads that we design, with optimization of the diagram of distribution of haul mass and rationalization of the costs of earth works (digging, filling, transport).

One of the created windows (Sketch) represents a view of each individual perspective, with preparations for printing. Prior to printing, the user defines the sheet sets, with sheets that are completely independent. This offers numerous possibilities in the creation of the appearance of the printed final forest road project.

Though the name of the interface (offered options and menus) is not a critical factor in the selection of the most appropriate tool, the user's understanding of the application has an important role in the selection of the computer program used. In the case of development of this computer program, the selection of the platform was based on the possibilities of internationalization and localization (WxLOcale klasa, WxGetTranslation), i.e. the ability to create an interface based on the user's requirements.

The newly developed computer program is specific in its openness to the addition of new module elements, which would unite all the phases in the creation of new forest roads.

The new computer program can be supplemented with new functions at any time, and the addition of new options is left as an open possibility and fits in well with the programming structure. The program development is based on its stability and the possibility of execution on all known operating systems. Ready-made functions were not implemented, instead, each is written specifically for the needs of this program. The majority of other computer programs use ready-made engines that enable movement through space, rendering, visualisation of the design, etc. The reason for this is the exceptional complexity of developing new engines, and there are many satisfactory solutions on the market.

5 Conclusions

The majority of existing computer programs (analysis computer programs) are not specialised for the design of forest roads. They offer many possibilities which are, most often, fully exploitable only in the design of public roads. As such, widely applicable and general computer programs are only partially useful in the design of forest roads.

The design of forest roads, in comparison with the design of public roads, has many specificities that are not considered in the design of computer programs intended for the design of public roads. Therefore, irresolvable issues often arise for designers trying to resolve specific forestry issues when using general road design programs.

The computer program CESTA created by the Slovenian company SoftData is the most commonly used program in Croatian forest road building. This program, however, was also primarily intended for the design of public roads, and subsequent adaptations in the design of forest roads (with the advice, instructions and guidelines of forestry experts). As such, it represents an acceptable transitional solution, but in no way represents an ultimate and long-term solution for Croatian forestry sciences.

It is certainly necessary to develop a theoretical model and, based on this, a concrete and specialised computer program that can be used to design various categories of forest roads, and would consist of a large number of modules. In addition to the modules for design, the program would also contain a planning modules and maintenance module for forest roads.

The newly designed computer program for designing forest roads is an original Croatian product, that in the future should be used as the official program for the design of all forest roads in the Republic of Croatia that require project documentation (today these are required only for primary roads, however, it is very likely that this will also be required for secondary forest roads in the future).

The same computer program will be used to educate the students of the Faculty of Forestry in group courses dealing with the issues relating to forest roads in undergraduate, graduate and postgraduate study programs, and life-long education of forestry experts working with forest roads (via the Croatian Chamber of Forestry Engineers and Wood Technologists).

The computer program is adapted for work on virtually all platforms, and is open to diverse challenges, ideas and needs of the forestry profession. There are the possibilities of expanding the computer program by adding specially designed modules (modules for planning primary forest roads, modules for planning secondary forest roads, modules for forest road maintenance, etc.).

The computer program will ensure the development of better quality and more standardized forest road projects, which are then easier to compare and allow for improved control. With the introduction of clear rules, this will enable the resolution of dilemmas that exist, and will make it impossible to deviate from the set framework.

It is assumed that better quality projects (created with the new computer program) will be a guarantee, with proper, full-time supervision, of the construction of high quality roads that will cost less to maintain.

These uniform main forest road projects, once archived in the database of forest road projects (at the level of the company Croatian Forests Zagreb Ltd. or, even better, at the level of the forest of all forest concession holders in the country), will form the basis for high quality and regular additions to the existing cadastre of primary forest road infrastructure.

6 Literature

ANSI (1995), Working Paper for Draft Proposed International Standard for Information Systems – Programming Language C++, Technical Report X3JI6/95-0087, American National Standards Institute (ANSI), April 1995.

Barton, J., J., Nackman, L., R. (1994), Scientific and Engineering C++ – An Introduction with Advanced Techniques and Examples, AddisonnWesley, Reading, MA, ISBN 0-201-53393-6.

BORLAND (1994), C++ 4.5 Programmer's Guide, Borland International, Scotts Valley, CA.

Carroll, M.D., Ellis, M.A. (1995), Designing and Coding Reusable C++, Addison-Wesley, Reading, MA, ISBN 0-201-51284-X.

Eckel, B. (1999), Thinking in C++, Prentice Hall, Englewood Cliffs, NJ, ISBN 0-13-917709-4.

Ellis, M., A., Stroustrup, B. (1990), The Annotated C++ Reference Manual, Addison-Wesley, Reading, MA, ISBN 0-201-51459-1.

Hanly, J., R., Koffman, E., B., Horvath, J., C. (1994), C Program Design for Engineers, Addison-Wesley, Reading, MA, ISBN 059064-6.

Horstmann, C., S. (1996), Mastering C++ – An Introduction to C++ and Object-Oriented Programming for C and Pascal Programmers (2nd Edition), John Wiley and Sons, New York, ISBN 0-471-10427-2.

Josuttis, N., M. (1999), The C++ Standard Library – A Tutorial and Handbook, Addison Wesley Longman, Reading MA, ISBN 00201-37926-0.

Kernighan, B., Ritchie, D. (1988), The C Programming Language (2nd Edition), Prentice-Hall, Englewood Cliffs, NJ, ISBN 0937699-2.

Lippman, S., B. (2000), Essential C++, Addison-Wesley Longman, Reading, MA, ISBN 0-201-48518-4.

Meyers, S. (1998), Effective C++ – 50 Specific Ways to Improve Your Programs and Design (2nd Edition), Addison-Wesley Longman, Reading, MA, ISBN 0-201-92488-9.

Nevečerel, H. (2010), Teoretical model and development of computer programme for designing of forest roads, Dissertation thesis, p. 233.

Pentek, T. (2010), Lectures, pptx.

Pentek, T., Nevečerel, H., Pičman, D., Poršinsky, T. (2007), Forest road Network in the Republic of Croatia - Position and Perspectives. Croatian Journal of Forest Engineering. 28 (2007), 1; p. 93-106.

Ryan, T., Phillips, H., Ramsay, J., Dempsey, J. (2004), Forest Road Manual - Guidelines for the design, construction and management of forest roads, COFORD 2004, p. 156.

Stroustrup, B. (2000), The C++ Programming Language (Special Edition), Addison-Wesley Longman, Reading, MA, ISBN 070073-5.