

Library of graphic symbols for power equipment in the scalable vector graphics format

A.G. Yuferov

Obninsk Institute for Nuclear Power Engineering, NRNU MEPhI, 1 Studgorodok, Obninsk, Kaluga Region 249040, Russia

Available online 6 May 2016

Abstract

This paper describes the results of developing and using a library of graphic symbols for components of power equipment under the state standards GOST 21.403-80 “Power Equipment” and GOST 2.789-74 “Heat Exchangers”. The library is implemented in the SVG (Scalable Vector Graphics) format. The obtained solutions are in line with the well-known studies on creating libraries of parametrical fragments of symbols for elements of diagrams and drawings in design systems for various industrial applications. The SVG format is intended for use in web applications, so the creation of SVG codes for power equipment under GOST 21.403-80 and GOST 2.789-74 is an essential stage in the development of web programs for the thermodynamic optimization of power plants. One of the major arguments in favor of the SVG format is that it can be integrated with codes. So, in process control systems developed based on a web platform, scalable vector graphics provides for a dynamic user interface, functionality of mimic panels and changeability of their components depending on the availability and status of equipment. An important reason for the acquisition and use of the SVG format is also that it is becoming the basis (recommended for the time being, and mandatory in future) for electronic document management in the sphere of design documentation as part of international efforts to standardize and harmonize data exchange formats. In a specific context, the effectiveness of the SVG format for the power equipment arrangement has been shown. The library is intended for solution of specific production problems involving an analysis of the power plant thermal circuits and in training of power engineering students. The library and related materials are publicly available through the Internet. A number of proposals on the future evolution of the library have been formulated.

Copyright © 2016, National Research Nuclear University MEPhI (Moscow Engineering Physics Institute). Production and hosting by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Keywords: CAD; Power equipment; Scalable vector graphics; SVG; Data exchange.

Introduction

Nowadays, the evolution of computer-aided design systems follows a “text+code” paradigm, which suggests textual representation of drawings and its integration with codes for design computations. Such approach eliminates a great deal of bottlenecks resulting both from the conventional paper-based technology for the handling of design documentation, and from computerization of design activities [1]. This also offers a number of opportunities:

- handling of a drawing as of a database containing graphic, numerical and textual elements;
- validity, consistency and verifiability of design documentation through the application of verification software to the drawing database;
- adaptability of drawings, repeated use thereof and management of the libraries of drawings (this minimizes the labor input in drafting, provides for the integrity of documentation, and simplifies its support and modification);
- parameterization, scaling, automatic display of dimensions and design parameters (weights, volumes, etc.) on the drawing using, for instance, DOM (*Document Object Model*), a standard open technology;
- unification of mimic panels and drafting graphic elements, automatic generation of installation diagrams based on schematics, etc.;

E-mail address: anatoliy.yuferov@mail.ru.

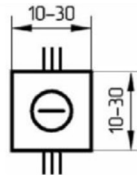
Peer-review under responsibility of National Research Nuclear University MEPhI (Moscow Engineering Physics Institute).

Russian text published: *Izvestia Visshikh Uchebnikh Zavedeniy. Yadernaya Energetika* (ISSN 0204-3327), 2015, n.4, pp. 79-89.

<http://dx.doi.org/10.1016/j.nucet.2016.03.013>

2452-3038/Copyright © 2016, National Research Nuclear University MEPhI (Moscow Engineering Physics Institute). Production and hosting by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

- automatic generation or updating of design documentation based on results of optimization design analyses;
- support, as part of a unified technology and a unified language environment, of all product description levels at all stages of the product lifecycle (schematic, assembly drawing, installation drawing), as well as of all types of design and operating documentation using, for instance, GOST R ISO 10303 standards (STEP or *STandard for Exchange of Product model data*), which include the EXPRESS object-oriented language implementing the EDI/XML (*Electronic Data Interchange/ Extensible Markup Language*) technology for the product model description at different stages of the product lifecycle;
- use of a dynamic graphic interface in automatic process control systems, functionalization of mimic panels, and modification of the composition (configuration) thereof depending on the availability and status of equipment components.

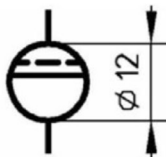


```

<svg width="20cm" height="15cm" version="1.1"
xmlns="http://www.w3.org/2000/svg"
style="fill:none; stroke: blue; stroke-width:.05cm; font-
family:Arial;">
<text x="1cm" y="3cm" font-size="15" fill="blue"> 2.1.
Reactor with the designation number of loops </text>
<rect x="5cm" y="5cm" width="2cm" height="2cm"/>
<line id="L1" y1="5cm" x1="5.75cm" y2="4cm" x2="5.75cm"/>
<line id="L2" y1="5cm" x1="6cm" y2="4cm" x2="6cm" />
<line id="L3" y1="5cm" x1="6.25cm" y2="4cm" x2="6.25cm"/>
<line id="L4" y1="7cm" x1="5.75cm" y2="8cm" x2="5.75cm"/>
<line id="L5" y1="7cm" x1="6cm" y2="8cm" x2="6cm"/>
<line id="L6" y1="7cm" x1="6.25cm" y2="8cm" x2="6.25cm"/>
<circle cx="6cm" cy="6cm" r="0.5cm"/>
<line id="svg_6" y1="6cm" x1="5.6cm" y2="6cm" x2="6.4cm"/>
<rect x=".01cm" y=".01cm" width="19.98cm" height="10.98cm"/>
</svg>

```

Fig. 1. A reactor with designation of the number of loops.



```

<svg width="20cm" height="15cm" version="1.1"
xmlns=http://www.w3.org/2000/svg
style="fill:none; stroke: blue; stroke-width:.05cm;>
<text x="1cm" y="3cm" font-family="Verdana" font-
size="30" fill="blue"> 2.9. Separator RBMK </text>
<line id="L2" y1="5cm" x1="6cm" y2="4.5cm" x2="6cm"/>
<line id="L5" y1="7cm" x1="6cm" y2="7.5cm" x2="6cm"/>
<circle cx="6cm" cy="6cm" r="1.0cm"/>
<line y1="5.4cm" x1="5.2cm" y2="5.4cm" x2="6.8cm" style
= "stroke-dasharray: 13, 6;"/>
<line y1="5.6cm" x1="5.1cm" y2="5.6cm" x2="6.9cm"/>
<rect x=".01cm" y=".01cm" width="19.98cm"
height="10.98cm"/>
</svg>

```

Fig. 2. A separator of an RBMK reactor.

Prospects of the SVG format in CAD

The first step towards the solution of the above problems is to use a text format for representation of drawings. The SVG format is a perfect choice for this [2,3]. Quite naturally, the existing drawing formats are also more or less capable to ensure that the above requirements are met. However, what makes the SVG format the preferred option is its simplicity, openness, extensive supportability, and compatibility with

other standards and, especially important today, with web-based technologies. This is where the SVG format is advantageous to the well-known CAD file formats, which (for the reasons explained by the creation and development history of these formats) require nowadays a great deal of efforts for the integration of drawings with external codes and with the DOM-technology, or for the remote editing of drawings.

A still more important reason for the acquisition and use of the SVG format is that it is gradually becoming the basis

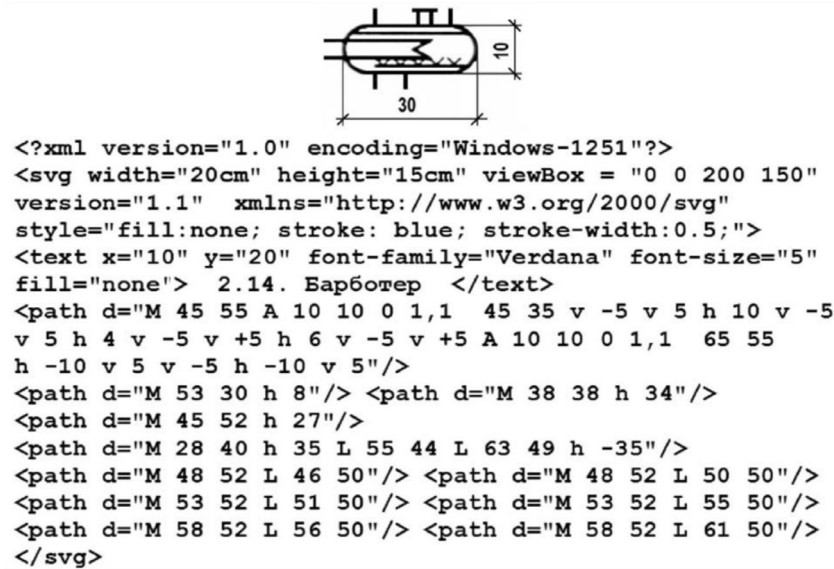


Fig. 3. A bubbler tank.

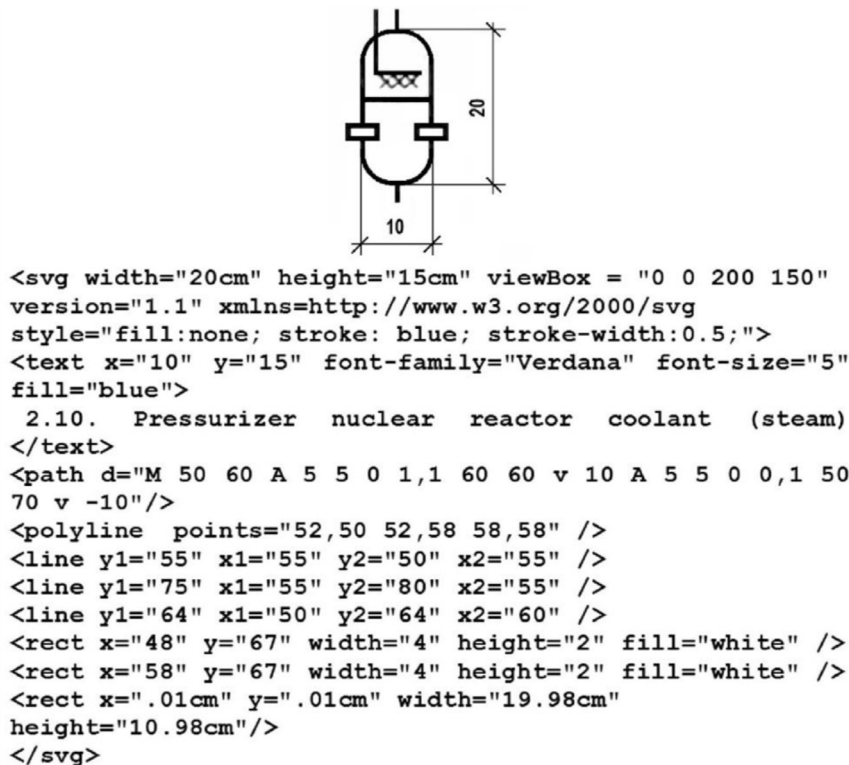
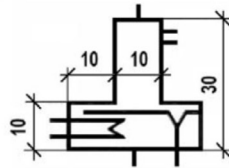


Fig. 4. A nuclear reactor coolant pressurizer (steam).



```
<?xml version="1.0" encoding="Windows-1251"?>
<svg width="20cm" height="15cm" viewBox = "0 0 200 150"
version="1.1" xmlns=http://www.w3.org/2000/svg
style="fill:none; stroke: blue; stroke-width:0.5;">
<text x="80" y="20" font-family="Verdana" font-size="5"
fill="blue"> 2.15. Деаэра́тор I конту́ра </text>
<path d="M 40 35 h 10 v -20 h 5 v -5 v 5 h 5 v 2.5 h 5 h
-5 v 2.5 h 5 h -5 v 15 h 10 v 20 h -30 v -20"/>
<path d="M 28 40 h 35 L 55 44 L 63 49 h -35"/>
<path d="M 45 38 h 19 L 66 43 v 15 v -15 L 68 38 h 2"/>
<path d="M 55 55 v 5" />
</svg>
```

Fig. 5. A primary circuit deaerator.

(recommended for the time being, and mandatory in future) for electronic document circulation in the sphere of design documentation for different industries. This is explained by an obvious need for standard unified formats for the international exchange of data, specifically of drawing data.

Unlike other graphic formats, the SVG format is fairly simple to import into codes for dynamic representation of numerical information.

The paper describes the implementation of graphic symbols for power equipment, as specified in the state standards GOST 21.403-80 “Power Equipment” and GOST 2.789-74 “Heat Exchangers” and used in a number of studies on optimization of power equipment [4–13]. The motivation for the creation of the library described herein was to provide graphic tools for the web-based analogs of well-known programs for thermodynamic optimization of power plants [14–16].

SVG codes for reactor equipment

SVG codes of graphic symbols for some of the reactor equipment components, as specified in GOST 21.403-80, are presented as examples of the SVG format application: a reactor with the designation of the number of loops (Fig. 1); a separator of an RBMK reactor (Fig. 2); a bubbler tank (Fig. 3); a nuclear reactor coolant steam pressurizer (Fig. 4); a primary circuit deaerator (Fig. 5). The codes for the rest of the graphic symbols, as specified in GOST 21.403-80 and GOST 2.789-74, are available on the described library’s website.

Example of the drawing layout

The SVG file shown in Fig. 6 presents tools for the thermal circuit arrangement using the available graphic symbols. The respective drawing is shown in Fig. 7.

A graphic symbol is recorded as an individual object with its elements contained in the tags of the group `<g>` `</g>` with the indication of the appropriate group name. The set of such graphic symbols required for the creation of a particular

drawing is stored in the SVG file, enclosed within the `<defs>` `</defs>` tags. This makes the set invisible, but makes it possible to select graphic symbols from the set according to their names and then copy the symbols in the required quantities to insert them into the desired drawing positions. For example, using this approach, the VVER primary circuit diagram can be described in three lines:

```
<use xlink:href="#PWR" x="30" y="10"/>
<use xlink:href="#GCN" x="45" y="55" />
<use xlink:href="#PWR_steam_generator" x="110"
y="-5" />
```

The SVG code presented in Fig. 6 is integrated with the computational module simulating deviations of the steam generator outlet temperature and pressure from rated values. The display of the current calculated values is enabled via periodic programmed refreshment of the web page into which the SVG code is embedded.

Once created, an SVG drawing may be repeatedly improved in different aspects and complemented with details and scenarios, or may be recorded as an independent object so that it could be inserted into other drawings. The required manipulations are performed using a graphic editor or in the text of the respective SVG file programmatically or manually. This is where the SVG format has a major advantage over other vector formats which require a specific processing environment.

In practice, with a drawing of a particular category, e.g. a power plant thermal circuit diagram, the draftsman develops a specific marking defining the potential positions and connections of graphic elements (a mesh of graphic symbols). In the SVG technology, the generation of a graphic symbol mesh is trivial. With such a standard marking being available, complicated layouts may be generated using only a text editor to specify the grid cell and the graphic symbol to be positioned in it. The description of the required graphic symbol may be stored in an external (library) file. An example of such graphic symbol positioning is shown in Fig. 8.


```

<?xml version="1.0" encoding="Windows-1251" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.0//EN"
"http://www.w3.org/TR/2001/REC-SVG-20010904/DTD/svg10.dtd">
<head> <title>Первый контур ВВЭР</title>
<script>
function fresh() { location.reload(); }
setInterval("fresh()",500);
</script> </head>
<body><svg width="250mm" height="120mm" viewBox = "0 0 250
120" version="1.1" xmlns="http://www.w3.org/2000/svg"
xmlns:xlink="http://www.w3.org/1999/xlink"
onload="PWR_steam();"
style="fill: none; stroke: blue; stroke-width: .5;
font-family: Arial; font-size: 4; ">
<script> <![CDATA[ function PWR_steam() {
// температура и давление на выходе из парогенератора
T=270+Math.random()*10; P = 5 + Math.random();
T =T.toFixed(0); P =P.toFixed(1);
textNode = document.createTextNode(T + " C, "+ P + " МПа");
t_pg_ou=document.getElementById("t_pg_ou");
t_pg_ou.appendChild(textNode); t_pg_ou.removeChild(
t_pg_ou.children[1]); }]]> </script>
<text x="30" y="10" fill="blue" > Первый контур ВВЭР</text>
<defs> <g id = "PWR" >
<rect x="50" y="50" width="20" height="20"/>
<line y1="50" x1="60" y2="40" x2="60"/>
<line y1="70" x1="60" y2="80" x2="60"/>
<circle cx="60" cy="60" r="5" />
<line y1="60" x1="56" y2="60" x2="64"/></g>
<g id = "PWR_steam_generator" >
<text id = "t_pg_ou" x="62" y="48" > ОСЦИЛЛЯЦИИ: </text>
<line y1="50" x1="60" y2="45" x2="60"/>
<line y1="70" x1="60" y2="75" x2="60"/>
<line y1="53" x1="53" y2="53" x2="67" />
<circle cx="60" cy="60" r="10"/>
<path d="M -20 55 h 85 L 60 60 L 65 65 h -45 v 55 h -40"/></g>
<g id = "GCN" > <circle cx="45" cy="45" r="10" />
<text x="41" y="41" > ПЦН </text>
<rect x="55" y="40" width="10" height="10"/>
<circle cx="75" cy="45" r="5" />
<path d = "M 45 35 v -5"/> <path d = "M 45 55 v 5" />
<path d = "M 57 40 v -5"/> <path d = "M 57 50 v 5" />
<path d = "M 63 50 v 5" /> <path d = "M 65 45 h 5" />
</g> </defs>
<use xlink:href="#PWR" x="30" y="10"/>
<use xlink:href="#GCN" x="45" y="55" />
<use xlink:href="#PWR_steam_generator" x="110" y="-5" />
<text x="91" y="59" fill="blue" > 322С, 16 МПа </text>
<text x="91" y="84" fill="blue" > 289С </text>
</svg> </body> </html>

```

Fig. 6. SVG file for the simplified diagram of the VVER reactor primary circuit.

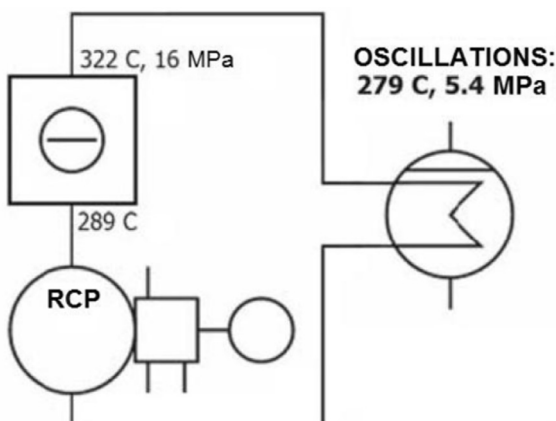


Fig. 7. An example of the drawing display in the browser.

For convenience, the graphic symbol connections may be specified on a separate finer mesh. Conspicuously, however, (see Fig. 8) certain calculations will be needed to avoid the grid cell displacements caused by the non-zero width of the marking lines. The SVG format makes it possible to incorporate the module for such calculations directly into the SVG file. Alternatively, measuring grids can be used, all dimensions of graphic symbols in which are specified in the grid spacings.

Conclusion

The results of the study are as follows.

1. A library of graphic symbols for components of power equipment, as specified in the state standards GOST

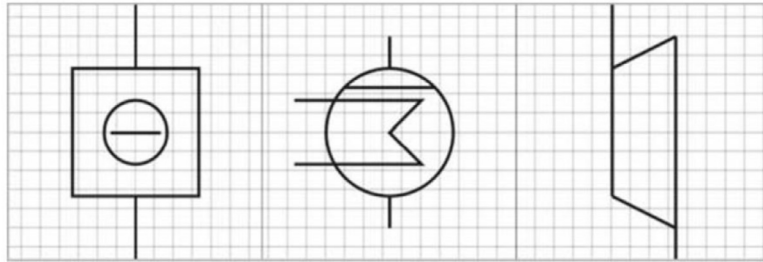


Fig. 8. An example of the graphic symbol positioning on the grid.

- 1.403-80 and GOST 2.789-74, was created in the SVG format.
2. A website has been created which allows free access to the library's current version. Individual drawings are available for preview at:
<http://178.215.91.20/eo>,
and the whole of the library is downloadable from:
<ftp://178.215.91.20/eo>.
3. The website contains materials with explanations on particular language structures used in the study to prepare SVG files and regarding the development of power equipment diagrams using SVG graphic symbols. Also, recommendations are provided for specifying the optimal display domain for the drawing in the browser, for specifying dimensions and for using Cyrillic fonts in SVG codes.
4. To simplify the arrangement of diagrams and drawings using standard elements, a concept of embedded scaling grids has been proposed.

Further objectives in the use of the SVG format for the description, visualization and design of power equipment diagrams are as follows.

1. Creating a library of graphic symbols modified with regard for the practical generation of power equipment diagrams. Specifically, the graphic symbol geometry needs to be improved as applied to their use on standard grids to simplify the scaling and creation of connections.
2. Creating animated (rotation, scaling, flood filling, dynamic text, etc.) graphic symbols for power equipment components to support design and drafting problems.
3. Adaptation of local and web-based SVG file editors for the generation of power plant diagrams, that is, accumulation of templates and scenarios in the respective editor's environment.
4. Integration with factual databases of power plants.
5. Integration of GOST R ISO 10303 with the EXPRESS language to enable description of product models. Both technologies are based on the XML language, which facilitates the task.
6. In future, such standards as GOST 21.403-80 and GOST 2.789-74, which specify the type of some graphic elements, shall be supported by respective computer codes of these elements in different formats.

References

- [1] D. Horafas, S. Legg, *Engineering Databases*, Mashinostroyeniye Publ., Moscow, 1990 (in Russian).
- [2] Scalable Vector Graphics (SVG) 1.1, second ed. W3C Recommendation. Available at: <http://www.w3.org/TR/2011/REC/SVG11/20110816>.
- [3] Kariyev Ch. *Masshtabiruyemaya Vektornaya Grafika* [Scalable Vector Graphics]. Available at: <http://intuit.ru/studies/courses/1063/210/info> (in Russian).
- [4] GOST 21.403-80. ESKD. Oboznacheniya Uslovnye Graficheskoye V Shemakh. Oborudovaniye Energeticheskoye [GOST 21.403-80. ESKD. Graphic Symbols in Diagrams. Power Equipment] (in Russian).
- [5] GOST 2.701-84. ESKD. Pravila Vypolneniya Skhem [GOST 2.701_84. ESKD. Rules for Generation of Diagrams] (in Russian).
- [6] GOST 2.789-74. ESKD. Oboznacheniya Uslovnye Graficheskoye. Apparaty Teploobmennyye [GOST 2.789-74. ESKD. Graphic Symbols. Heat Exchangers] (in Russian).
- [7] I.V. Gordeyeva, V.N. Kaurkin, N.G. Mironova, V.N. Netunayeva, T.A. Poltavtseva, Yu.V. Stepanov, G.M. Frolova, *Generation of Thermal Circuit Diagram for Power Plants*, MEI Publ., Moscow, 2010 (in Russian).
- [8] G.B. Levental, L.S. Popyrin, *Optimization of Heat Power Plants*, Energiya Publ., Moscow, 1970 (in Russian).
- [9] L.S. Popyrin, *Mathematical Modeling and Optimization of Heat Power Plants*, Energiya Publ., Moscow, 1978.
- [10] L.S. Popyrin, *Mathematical Modeling and Optimization of Nuclear Power Plants*, Nauka Publ., Moscow, 1984 (in Russian).
- [11] L.S. Popyrin, V.I. Samusev, V.V. Epelshteyn, *Automation of Mathematical Modeling of Heat Power Plants*, Nauka Publ., Moscow, 1981 (in Russian).
- [12] A.A. Kanaev, E.F. Ratnikov, I.Z. Kopp, *Thermodynamic Cycles, Diagrams and Equipment of Nuclear Power Plants*, Atomizdat Publ., Moscow, 1976 (in Russian).
- [13] D.P. Gokhshteyn, G.P. Verkhivker, *Analysis of Thermal Circuit Diagrams for Nuclear Power Plants*, Vishcha shkola Publ., Kiev, 1977 (in Russian).
- [14] A.V. Moshkarin, B.L. Shelygin, G.I. Doverman, Ye.V. Zakharenkov, T.A. Zhamlikhanov, Ye.S. Malkov, *Vestnik IGEU* (4) (2011) 6–10 (in Russian).
- [15] D.K. Smirnov, N.N. Galashov, *Izv. Tomsk. Politekh. Univ.* 320 (4) (2012) 6–10 (in Russian).
- [16] M.A. Nikolayev, *Nauchno-tehnicheskoy Vestnik Povolzhya* (1) (2014) 130–133 (in Russian).