“POLITEHNICA” UNIVERSITY of BUCHAREST
DOCTORAL SCHOOL OF INDUSTRIAL AND ROBOTIC ENGINEERING
Decision no. 762 of 19.10.2021

SUMMARY
PhD. THESIS

GROUP DECISIONS SYSTEM SUPPORT IN MECHANISMS DESIGN

PhD. candidate: **Eng. Moșneaga Anca-Olimpia (Talaat-Hamid)**

PhD supervisor: **Distinghuised Professor PhD. Eng. Iosif Tempea**

PhD. COMMISSION

Chairman	**Professor PhD. Eng. Nicolae Ionescu**	from	**POLITEHNICA University of Bucharest**
PhD supervisor	**Distinghuised Professor PhD. Eng. Iosif Tempea**	from	**POLITEHNICA University of Bucharest**
Reviewer	**PhD. Associate professor Eng. Amelitta Legendi**	from	**Technical University of Constructions of Bucharest**
Reviewer	**Professor PhD. Eng. Nicolae Dumitru**	from	**University of Craiova**
Reviewer	**Professor PhD. Eng. George Adîr**	from	**POLITEHNICA University of Bucharest**

BUCHAREST
2021

Contents

I want to take this opportunity to especially thank mr. Distinguished Professor PhD. Eng. Iosif Tempea, PhD supervisor, for the support and guidance he gave me during my doctoral period and for all the prompt, precise and competent support I received.

Special thanks to Professor PhD. Eng. George Adîr and Professor PhD. Eng. Constantin Ocnărescu, for the support received throughout the period necessary to complete the thesis.

I would like to thank the reviewers of this PhD. thesis, for their patience and attention regarding the analysis of the doctoral thesis.

I would like to thank all the teachers in the Department of Theory of Mechanisms and Robots of the POLITEHNICA University of Bucharest, for the help and support received.

I would like to especially thank my mother, Moşneaga Olimpia-Valeria, and my children, Laura and Andu, who supported me and were close to me during this period.

The paper "Support system for group decisions in the design of mechanisms" is the synthesis of research conducted during the doctoral and master's years, spent at the POLITEHNICA University of Bucharest, Faculty of Industrial and Robotics Engineering. During these years I gained a unique professional, but also a life experience.

Author

CHAPTER I

FOREWORD

The prosperity of any economic structure is largely conditioned by the efficiency of the decisions taken by managers at each hierarchical level of management. Thus, the importance of choosing and transposing decisions into life is undeniable, depending on the environment in which the economic object operates and the conditions imposed by its technological and organizational level.

Following the undertaken research, it can be stated that there are a small number of support systems for group decisions in the field of mechanism design, based on Romanian products. Some companies provide tools for selecting products based on a very limited number of criteria.

The bibliographic study carried out did not highlight a modern software product, implemented in a state-of-the-art programming language, which would provide support to specialists, both in the field of education and in the industrial field.

Through this paper, all the stages of designing a support system for group decisions in the design of mechanisms are analyzed. From the multitude of components used in the process of designing the mechanisms, two components were chosen for which the related expert system was designed, as a component part of the support system for group decisions.

CHAPTER II

ARTIFICIAL INTELLIGENCE

Artificial intelligence can be defined as the field of computer science that develops technical systems capable of solving difficult problems related to human intelligence”[1], simulating human intelligence through the computer.

The calculation system based on artificial intelligence involves solving problems based on reasoning, but also by “learning” from the situations presented before the system or from the interaction with the external environment.

There is a direct link between information and intelligence. Mankind is characterized by intelligence but the possibilities of knowledge are limited. To address this shortcoming, database management systems are being used to store a huge amount of information, at low cost and with high performance. At the same time, the stored information can be combined and extracted depending on the specific purposes of the problem to be solved.

The multitude of scientific concerns in the field of robotics has highlighted the many achievements in this field, which support the widest possible development of the introduction of large-scale artificial intelligence.

CHAPTER III
EXPERT SYSTEMS

3.1. FOREWORD

One of the areas in which artificial intelligence is mainly applied is the field of expert systems. Expert systems can be considered as tools with which the human factor can solve various problems in a specific field. Basically, they are made up of programs that incorporate the knowledge of human experts in a particular field and programs that simulate the reasoning of these experts in determining optimal solutions based on stored knowledge.

Unlike human experts, expert systems have several limitations: first, the computer has no intuition. A human expert, based on his knowledge and experience, will choose the optimal solution to solve a problem using intuition. On the other hand, even if many of the production rules in a field are implemented, they are not exhaustive. The number of analogies that an expert system can make, as well as the factors that it takes into account when choosing a solution, are limited due to this limited baggage of knowledge that the expert system receives. For this reason, the expert system is an extremely useful but limited tool. However, with all these limitations, compared to a human expert, the expert system will execute inferences at a much higher speed.

3.2. THE ARCHITECTURE OF THE EXPERT SYSTEM

The expert system has the following components:

- Knowledge base,
- Facts,
- The mechanism of inferences.

The design, execution and implementation of an expert system requires a close collaboration between the human expert, specialist in the specified field, and the programming engineer who deals with the software part of the system. The human expert will pass on his knowledge, acquired through his experience, to the programming engineer. The form in which this knowledge is transmitted must be as clear as possible. In addition, in order to ensure the highest possible accuracy of the results obtained by the expert system, the knowledge made available to the programmer must be as complete as possible. If the facts and knowledge of the human specialist can be proved by science, heuristic knowledge is the result of experiences, intuition or partially demonstrable reasoning.

In general, in addition to these three modules, an expert system may also contain: the user interface module, the knowledge acquisition module and the explanation module.

The knowledge base is designed as a module for storing the knowledge received from the human expert. The probability that the solution determined by the expert system is correct depends on the correctness and completeness of this module. The organization of this knowledge base will determine the speed of response of the system, as the knowledge will be selected according to certain criteria, according to the problem to be solved. If the form of knowledge storage also includes rules, then in the knowledge base there is also the fact base and the rule base.

A most important component of the expert system is the knowledge acquisition module. Through it, the knowledge in the knowledge base, as well as the rules in the rules base, can be updated. The task of the programming engineer is extremely difficult: the performance of the expert system depnds on the way in which he manages to design and organize this knowledge base.

The inference mechanism accesses the knowledge from the database and processes it according to a solving algorithm established according to the type of problem to be solved and the input data provided by the user. There are several solving strategies: forward (deductive) and backward (inductive).

The fact base is a separate, knowledge-based module and contains input data and intermediate data obtained from the application of inferences according to the problem-solving algorithm.

The knowledge acquisition module is intended for inserting, modifying, deleting knowledge from the knowledge base. Through this module the knowledge base is created and updated. The programming engineer, through a close collaboration with the human expert, designs and organizes the knowledge base.

The module that provides the user interface offers him the possibility to provide input data, as well as the restrictions imposed. A friendly, clear and easy to use interface will make it easy for the user to access the expert system.

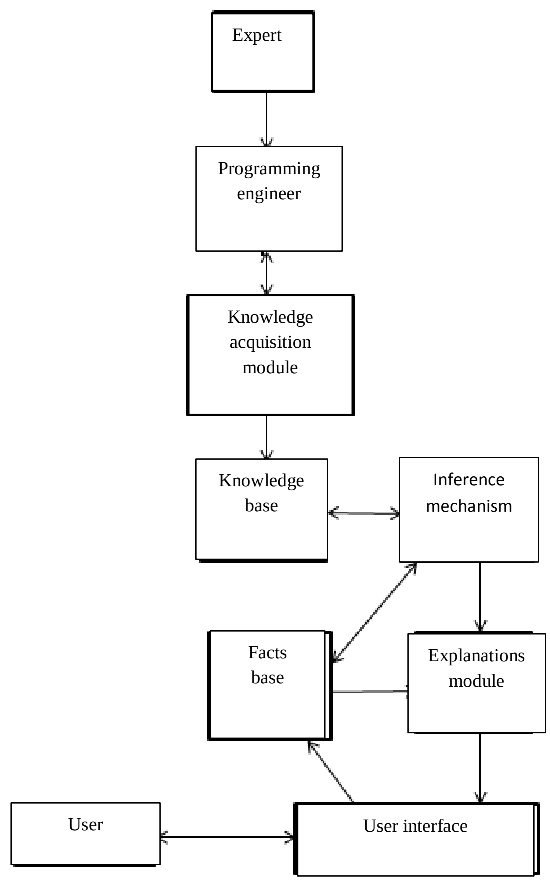


Fig.3.2. Expert system architecture

CHAPTER IV IV

THE DECISION AND THE DECISION-MAKING PROCESS

Decisions involve a rational choice between several options. This chapter presents the forms of the decision, the modeling of the decision-making processes, the classifications and characteristics of the decisions.

Decision-making is today an increasingly complex and difficult process, due to the increase in the amount of information and fluctuations in the economic environment. For this reason, decision-makers need technical assistance to help them make quality decisions in a short period of time, and this assistance is provided through the use of a decision support system.

CHAPTER V

THE STRUCTURE OF THE EXPERT SYSTEMS

In this chapter are presented in detail the components of the expert system, their characteristics and the importance of each module in the operation of the expert system. In addition, aspects regarding the representation of knowledge in the knowledge base are analyzed.

Designing and organizing the knowledge base are essential components of the process of building an expert system. The programming engineer, in close collaboration with the human expert, must design as efficiently as possible the structure of the database tables for a correct and fast operation of the expert system. After the implementation of the expert system, its knowledge base must be able to be updated by insertions, modifications and deletions, in order to maintain the consistency of the data. The expert system simulates the reasoning of the human expert using the knowledge taken from specialists in the field.

The representation of knowledge in the knowledge base considers three aspects: the inference mechanism, the language of THE knowledge representation and the field of expertise. Through the representation of knowledge it is desired to formalize and organize data and rules.

The inference mechanism follows the reasoning of the specialist in order to obtain from the input data the solution of the problem in optimal conditions.

The language of knowledge representation means how the data are retained in the knowledge base of the expert system.

The field of expertise represents the specialized field of knowledge, treated by the expert system.

CHAPTER VI
DECISION SUPPORT SYSTEM

The decision support system is: “a model based on a set of procedures for data processing and for assisting a manager in the decision-making process. The decision support system must be simple, robust, easy to maintain, adaptive, easy to communicate with, etc.”

This chapter presents the characteristics of the decision support system, their classification, specifying the place and importance of the group decision support system.

The form of organization called “group” involves several people who have common interests or skills, led by a leader. In the decision-maker group, the leader is the one who plans the activities necessary to make a group decision: meetings, debates, conflict resolution and reaching consensus. [67][98]

The group decision support system relies on computers to intensify group decision-making through information and communication technology, facilitating the exchange of information between group members and group interaction, in order to solve semi-structured or unstructured problems.

CHAPTER VII
GROUP DECISIONS SUPPORT SYSTEM

In order to design a support system for group decisions, it is necessary to establish the types of users of the system, as the proper conduct of the decision-making process in the case of group decisions depends to a large extent on resolving conflicts that may arise.

Decision support systems can be used by all company managers, regardless of their hierarchical level. Among the users of the system there can also be specialists who occasionally participate in the decision making, being experts in certain fields. Managers and specialists are the beneficiaries of the group decisions support system. In addition, for the entry of the necessary data, operators are considered users as well.

The group decision support system is based on an IT system. The programming engineer will collaborate with the managers to establish the design requirements, according to the concrete situations. Then, the programmer will model the processes that take place in the organization, building an information system corresponding to the technical and technological resources.

7.1. ESTABLISHING DESIGN REQUIREMENTS

În proiectarea unui sistem suport pentru decizii de grup trebuie să evidențiem diferențele procesului decizional de grup față de procesul decizional individual.

In designing a support system for group decisions we need to highlight the differences of the group decision-making process compared to the individual decision-making process.

The decision taken by a person is a decision independent of the values, knowledge, conclusions of other people. There are no conflicts in this situation, there are no negotiations or cultural norms to comply with, other than those of the decision maker. [60 [63]

In the case of group decisions, in order to make a decision, negotiations are used, conflicts are resolved, group work techniques are used, such as: the Delphi method, Brainstorming, the nominal group technique. In addition, the culture of the organization plays an important role in the decision-making process.

7.2. DESIGNING THE GROUP DECISIONS SUPPORT SYSTEM

The following requirements must be met when designing a group decision support system: [21][42]

- the decision support system is a tool that assists people in making a decision. It does not replace it.
- access to the system can be made from several locations,
- the system architecture can be changed whenever necessary,
- way of representing knowledge appropriate to the field,
- decentralization of decision making. The decision support system must not allow centrally controlled decisions.
- controlling the occurrence of conflicts and an effective mechanism for resolving them,
- easy, user-friendly interface,
- collaboration and integration between the various component modules of the decision support system.

7.3. EVALUATION OF GROUP DECISIONS SUPPORT SYSTEMS

In evaluating a group decision support system, emphasis is placed on its performance against the set objectives. Being a software product, the implementation of such a system is done by programming engineers.

The evaluation criteria of a group decisions support system are: [43] [54]

- the level of trust it has gained from operating the implemented program,
- the price of the product as new technologies are constantly emerging,
- adaptability to as many types of infrastructure as possible,
- data and user security,
- modularity and the possibility of product development by users,
- easy use by any person without specialist knowledge.

CHAPTER VIII

KNOWLEDGE BASE DESIGN

8.1. KNOWLEDGE REQUIRED FOR THE DESIGN OF BEARING MECHANISMS

The wide use of bearings, from heavy mining or construction machines to high-end equipment in the aeronautical industry or precision medical equipment, has led to a wide variety of construction technologies.

Research in the field of bearing design and construction aims to ensure their operation under the various conditions listed. Thus, there are a number of conditions that bearings must meet and sets of properties that they must possess, such as: [27] [53]

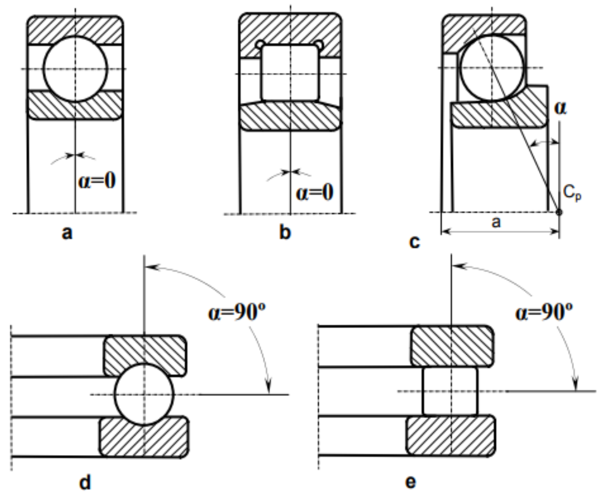


Fig.8.1. Contact angle of the bearings.
a) radially with balls in a row; b) radially with cylindrical rollers in a row; c) radially-axially with balls in a row, d) axially with balls in a row; e) axially with cylindrical rollers in a row; f) radially-oscillating with barrel rollers on two rows; g) radially-axially with conical rollers [27]

- the bearings must ensure the tightness of the equipment in all working conditions;
- operation at high temperatures;
- low working noise
- high durability;
- very high accuracy;
- various speeds;
- high durability in conditions of small volumes;
- high reliability as mounting conditions can be very difficult.

Extensively, knowledge about bearings can be synthesized using the following sets of attributes: [27] [82]

Set 1 = {radial force first bearing housing, radial force second bearing housing, axial force between first and second bearing housing, axial force between second bearing housing and first bearing housing} radial and axial forces;

Set 2 = {speed, speed limit} speeds;

Set 3 = {service life, durability} durability;

Set 4 = {shaft diameter in first bearing housing, housing bore diameter in first bearing housing, shaft diameter in second bearing housing, housing bore diameter in second bearing housing} dimensions of available volume;

Set 5 = {distance between bearings, angle of rotation in the first bearing housing, angle of rotation in the second bearing housing} shaft characteristics: length and rigidity;
Set 6 = {operating temperature, maximum temperature, minimum temperature, operating temperature} temperature characteristics;
Set 7 = {precision of movement, precision of manufacture} precision;
Set 8 = {limit noise produced during operation} noise level;
Set 9 = {humidity, level of impurities} environmental characteristics;
Set 10 = {required lubricant, lubrication system, designed assembly tightness level} lubrication and sealing

8.2. KNOWLEDGE REQUIRED TO DESIGN THE ROBOT OPERATION SYSTEM

The following tables will be created for choosing direct current motors: Motor, Motor size

Extensively, knowledge of direct current motors can be synthesized using the following sets of attributes [7] [80]:
Set 1 = {driver torque Tc, maximum torque Tpk, friction torque Tf} - torques;
Set 2 = {Snl idle speed} - speeds;
Set 3 = {motor inertia Jm,} - inertia;
Set 4 = {Wm weight, length, width} - dimensions and weight of the engine;
Set 5 = {motor constant Km, torque constant Kt, electrical constant and mechanical constant} - constants;
Set 6 = {working temperature} - temperature characteristics;
Set 7 = {infinite source impedance, zero source impedance} - impedances;
Set 8 = {operating voltage V} - electrical voltage;
Set 9 = {resistance Rt, inductance L} - total resistance and inductance;
Set 10 = {current intensity at idle, Inl, current intensity at maximum load Ip} - current intensity.

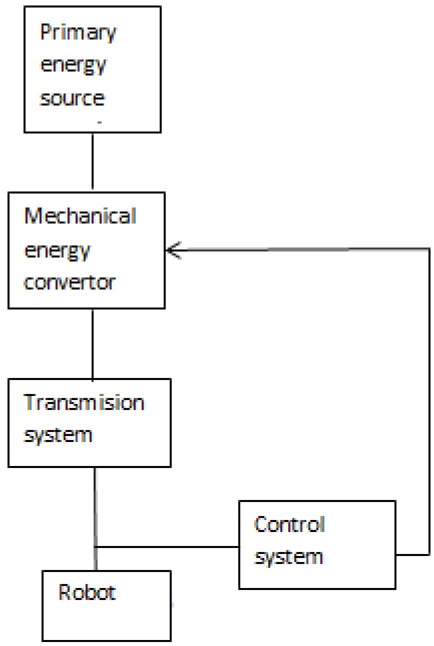


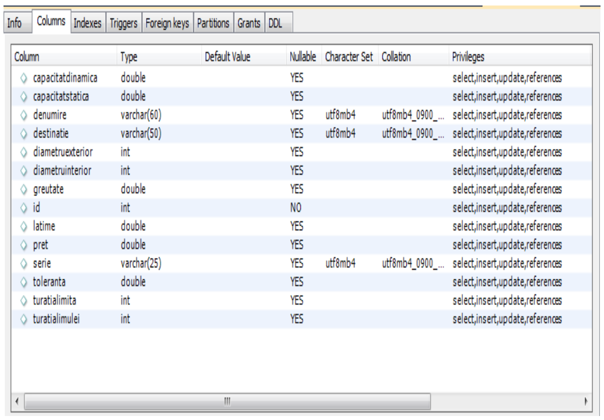
Fig. 8.2. General structure of an action system [5]

CHAPTER IX

DATABASE DESIGN

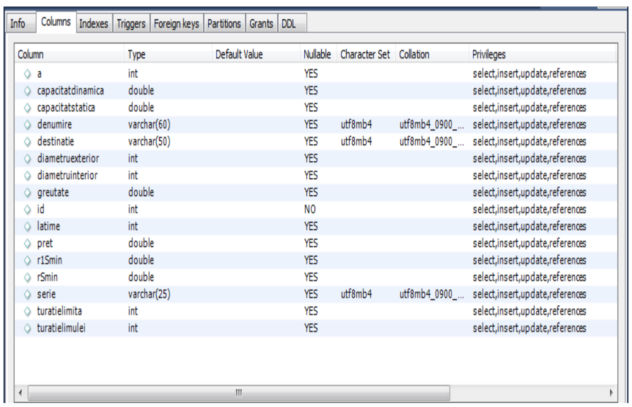
9.1. DESIGNING THE TABLES NECESSARY FOR THE CALCULATION AND VERIFICATION ALGORITHMS OF THE BEARINGS

The knowledge base, which contains data on bearings, consists of tables: Type, FaCo, FaCoradax Rulmenticubile, Rulmentiradaxcubile, Rulmentiradcurolecilindrice, Rulmentiradaxcuroleconice.



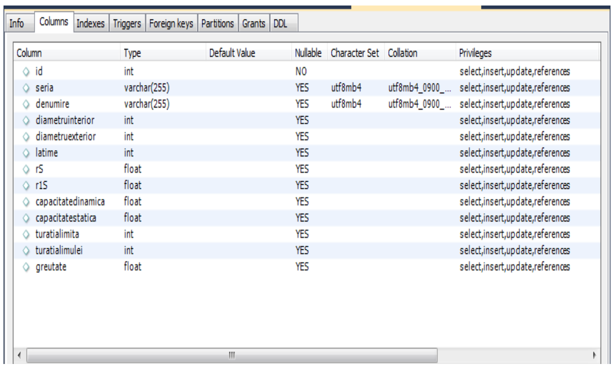
Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
capacitatdinamica	double		YES			select,insert,update,references
capacitatstatica	double		YES			select,insert,update,references
denumire	varchar(60)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
destinatie	varchar(50)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
diametruexterior	int		YES			select,insert,update,references
diametruinterior	int		YES			select,insert,update,references
greutate	double		YES			select,insert,update,references
id	int		NO			select,insert,update,references
latime	double		YES			select,insert,update,references
pret	double		YES			select,insert,update,references
serie	varchar(25)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
toleranta	double		YES			select,insert,update,references
turatialimita	int		YES			select,insert,update,references
turatialimulei	int		YES			select,insert,update,references

Fig.9.1. The structure of the Rulmenticubile table



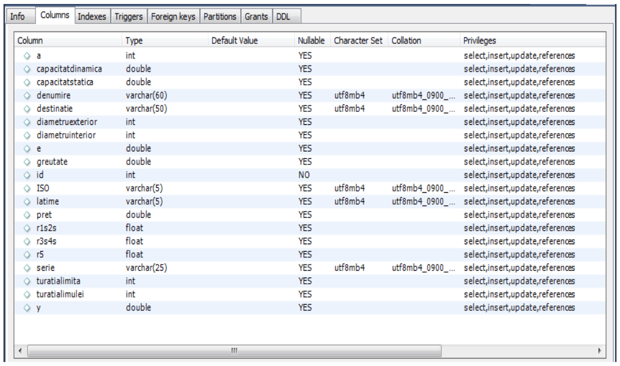
Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
a	int		YES			select,insert,update,references
capacitatdinamica	double		YES			select,insert,update,references
capacitatstatica	double		YES			select,insert,update,references
denumire	varchar(60)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
destinatie	varchar(50)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
diametruexterior	int		YES			select,insert,update,references
diametruinterior	int		YES			select,insert,update,references
greutate	double		YES			select,insert,update,references
id	int		NO			select,insert,update,references
latime	int		YES			select,insert,update,references
pret	double		YES			select,insert,update,references
r1Smin	double		YES			select,insert,update,references
rSmin	double		YES			select,insert,update,references
serie	varchar(25)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
turatielimita	int		YES			select,insert,update,references
turatielimulei	int		YES			select,insert,update,references

Fig.9.2**.** The structure of the Rulmentiradaxcubile table



Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
id	int		NO			select,insert,update,references
seria	varchar(255)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
denumire	varchar(255)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
diametruinterior	int		YES			select,insert,update,references
diametruexterior	int		YES			select,insert,update,references
latime	int		YES			select,insert,update,references
rS	float		YES			select,insert,update,references
r1S	float		YES			select,insert,update,references
capacitatedinamica	float		YES			select,insert,update,references
capacitatestatica	float		YES			select,insert,update,references
turatialimita	int		YES			select,insert,update,references
turatialimulei	int		YES			select,insert,update,references
greutate	float		YES			select,insert,update,references

Fig. 9.3 The structure of the Rulmentiradcurolecilindrice table



Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
a	int		YES			select,insert,update,references
capacitatdinamica	double		YES			select,insert,update,references
capacitatstatica	double		YES			select,insert,update,references
denumire	varchar(60)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
destinatie	varchar(50)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
diametruexterior	int		YES			select,insert,update,references
diametruinterior	int		YES			select,insert,update,references
e	double		YES			select,insert,update,references
greutate	double		YES			select,insert,update,references
id	int		NO			select,insert,update,references
ISO	varchar(5)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
latime	varchar(5)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
pret	double		YES			select,insert,update,references
r1s2s	float		YES			select,insert,update,references
r3s4s	float		YES			select,insert,update,references
r5	float		YES			select,insert,update,references
serie	varchar(25)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
turatialimita	int		YES			select,insert,update,references
turatialimulei	int		YES			select,insert,update,references
y	double		YES			select,insert,update,references

Fig.9.4. The structure of the Rulmentiradaxcuroleconice table

9.2. ENTERING THE DATA NEEDED TO DESIGN THE BEARINGS IN THE KNOWLEDGE BASE

Forms for each database table were created in a very "friendly" way. These forms provide the knowledge acquisition interface.

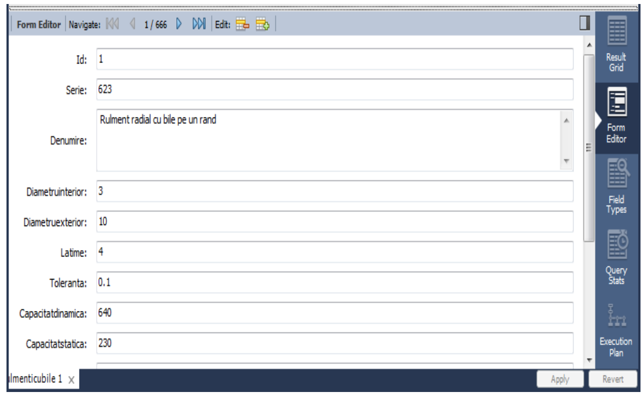
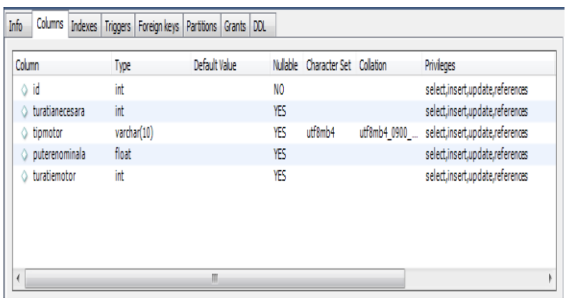


Fig.9.4. Data entry form

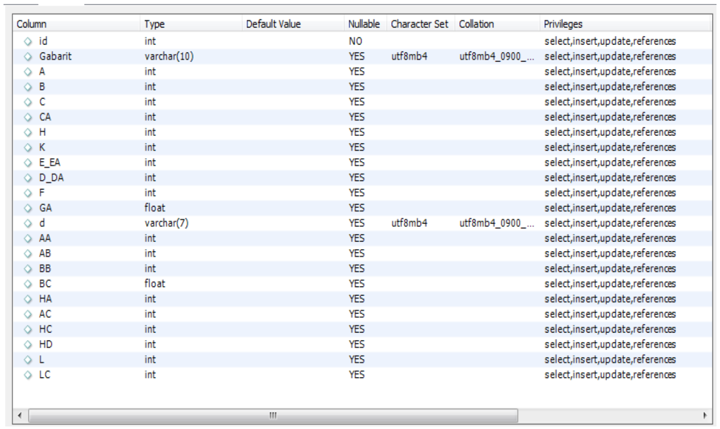
9.3. DESIGNING THE TABLES NECESSARY FOR THE DESIGN OF THE OPERATING SYSTEM

The knowledge base that contains data on direct current motors consists of the tables: Motor, Motor Dimension.



Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
id	int		NO			select,insert,update,references
turatianecesara	int		YES			select,insert,update,references
tipmotor	varchar(10)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
puterenominala	float		YES			select,insert,update,references
turatiemotor	int		YES			select,insert,update,references

Fig. 9.5**.** Engine table structure



Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
id	int		NO			select,insert,update,references
Gabarit	varchar(10)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
A	int		YES			select,insert,update,references
B	int		YES			select,insert,update,references
C	int		YES			select,insert,update,references
CA	int		YES			select,insert,update,references
H	int		YES			select,insert,update,references
K	int		YES			select,insert,update,references
E_EA	int		YES			select,insert,update,references
D_DA	int		YES			select,insert,update,references
F	int		YES			select,insert,update,references
GA	float		YES			select,insert,update,references
d	varchar(7)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
AA	int		YES			select,insert,update,references
AB	int		YES			select,insert,update,references
BB	int		YES			select,insert,update,references
BC	float		YES			select,insert,update,references
HA	int		YES			select,insert,update,references
AC	int		YES			select,insert,update,references
HC	int		YES			select,insert,update,references
HD	int		YES			select,insert,update,references
L	int		YES			select,insert,update,references
LC	int		YES			select,insert,update,references

Fig. 9.6**.** Engine Dimension table structure

Dimension knowledge is specified in the "Engine Dimensions" table based on the following characteristics:

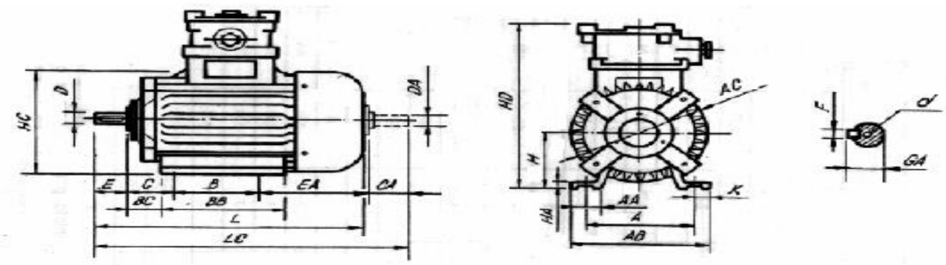


Fig. 9.7. Engine sizing [52]

9.4. ENTERING THE DATA NECESSARY FOR THE DESIGN OF THE KNOWLEDGE BASE OPERATING SYSTEM

Forms for each database table were created in a very “friendly” way. These forms provide the knowledge acquisition interface.

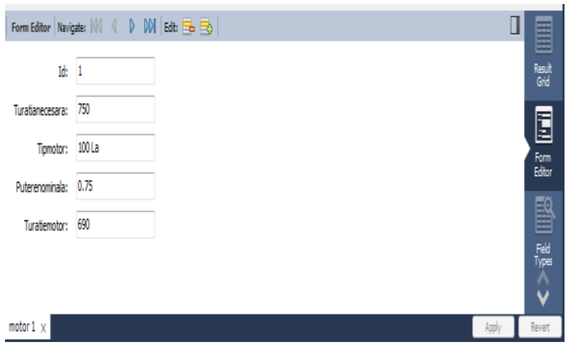


Fig. 9.8. Data entry form

9.5. ENSURING THE SECURITY OF THE APPLICATION

To ensure the security of the application we created a table in the database containing users and their login data.

Tabel 9.1. Structure of the Users table

Attribute name	Data type
name and surname	text
e-mail	text
Customer	text
password	text

This table will retain the identification data and login data of the application users. In this way, only users who are allowed access to the application will have access to the application, and through the facilities provided by MySQL, you can view all the information on the date and time when each user in the table accessed the database, as well as the tables accessed by them.

CHAPTER X

OPERATING ALGORITHMS

The strength of the decision support system is based on the ability of the inference mechanism to obtain intermediate data/solutions based on certain conditions. The inference mechanism must draw some conclusions or test some hypotheses by accessing the fact base, the knowledge base and the rules base. This is the problem solving strategy adopted and can be: deductive strategy or inductive strategy.

10.1. REPRESENTATION OF THE SOLUTION OF THE PROBLEM

The representation of the solution of the problem through the space of states consists of the triplet (S_i, O, S_f) where:

S_i is the set of initial states;
O is the set of operators that can be applied to the initial state. For a given state, there is a selection of operators that can be applied
S_f is the set of final states.

Therefore, a solution of the problem will determine the obtaining of a final state by successively applying some operators that transform the initial state into a final solution, passing through intermediate states.

As you search the space, part of the graph becomes explicit. This part of the graph that represents the search space is the traversed part of the search space.

10.3. SEARCH STRATEGIES

A search strategy based on traversing graphs in width or depth has a low control cost, but a higher cost of applying operators, i.e. traversing the graph from one node to one of its neighbors. It is generally necessary to go through a large number of states in order to determine the optimal path to the final state.

10.3.1. Search in width

Searching in width is a strategy that systematically examines accessible states in the initial Și state. This strategy uses a data structure called a “queue”, using a FIFO strategy. The current node, the one that expands and then is removed, is first in the list, and its successors are added to the end of the list. A one-dimensional array boolean s is used, where

s[i]= {

	1 if node I was visited
	0 if the node was not visited

Width search algorithm in the state space:

1. Grant access to queue tables[ ] , s[ ]
2. Insert the first new in the queue /* initial state */
3. Aslong as the queue[ ] is not void
4. Expand the S_c node /* current node */
   Examine all S_j direct successors S_c node
   For each S_j successor of S_c executes
   Apply the operator $S_j \Rightarrow S_c$
   If S_j is the final state then
   The solution is $(S_i, \ldots, S_j)$
   Returns SUCCESS /* solution is found */
   Otherwise
   Insert S_j node at the end of the queue
5. Eliminate the S_c node from the queue
6. If the final solution has not been reached then
   Returns INSUCCESS /* no solution found*/

The implementation in C ++ of this algorithm is listed in Annex 1.

10.3.2. Search in depth

In-depth search expands the most recently generated states. Therefore, this strategy traverses the graph from the initial state to the final state or, in the case of failure, to a state that has no successor. If no solution to the problem is determined, it is resumed by the backtracking method. This strategy uses a data structure called stack, st [ ], using the LIFO (Last In First Out) technique. [64] [90]

In the case of in-depth search, it cannot be guaranteed to determine a solution, as there is the possibility of going through a very large branch in depth, a branch that may not contain the solution to the problem. To avoid this situation, a maximum search depth limit, admax, is set. If this limit has been reached without finding the solution, then return and examine the neighboring state nodes with the state nodes on the levels below the maximum depth, but located on different branches. In this way, the solution that could be determined on the initial path at a depth greater than the maximum depth, is lost. In contrast, the depth search algorithm can traverse fewer nodes than the width search.

A one-dimensional array boolean s is used, where

s[i]= {
- 1 if node I was visited
- 0 if the node was not visited

Depth search algorithm in the state space:

1. Build table s [ ], t [ ], st [ ]
2. Inserts the initial state into the stack
3. As long as the stack is not empty then
4. For S_c node from the top of the stack
   If the top of the stack > admax then
   Eliminate the bode from the stack

5. Exapand the S_c node
   Examine the S_j neighbours of the S_c node
6. For each S_j neighbor of S_c execute
   Apply the operator $S_j \Rightarrow S_c$
   If S_j is the final state
   The solution is $(S_i, \ldots, S_j)$
   Return SUCCES /* solution is found */
   Otherwise
   Insert S_j on the top of the stack
   Update ad [j]

The implementation in C ++ of this algorithm is listed in Annex 2.

10.4. BEARING SIZING ALGORITHMS

The system offers the possibility of designing mechanisms that include the following types of bearings: radial with balls, radial-axial with balls, radial with cylindrical rollers and radial-axial with cyllindrical rollers. For each of these types of bearings, a calculation algorithm has been created using the deductive strategy: the user enters design data, the algorithm queries the database and selects in the facts base the bearings that correspond to the restrictions, then the list of these bearings is displayed. The selection on the basis of facts is made in two stages, first the bearings are selected according to the diameter of the shaft, then the selection is refined by means of design restrictions.

Finally, a data set is created containing information about the bearings that meet the requirements entered. The list is displayed to the user, who selects a bearing from the list to view its parameters. After analyzing the data, the user can choose to save them in a file for later printing or can resume the algorithm for calculating a new bearing.

10.5. BEARING VERIFICATION ALGORITHM

The algorithm uses both the inductive strategy and the deductive strategy to verify the correctness of the choice of a bearing, as a component part of the designed mechanism. The application offers the possibility to design mechanisms that include the following types of bearings: radial with balls, radial-axial with balls, radial with cylindrical rollers and radial-axial with conical rollers.

The user is invited to enter the design data specific to the bearing to be checked: bearing series, shaft diameter, speed, radial force, axial force, permissible operating life. Depending on the type of bearing, the database is queried and the bearings corresponding to the entered data are saved in the fact base. Thus, the user can check if the bearing has been well dimensioned and can also view the series of bearings with which it can be replaced within the designed mechanism. The user selects a bearing from the list, not necessarily the one to be checked, to list the information about it. The selection from the database of bearings that can be used instead of the bearing subject to verification is performed in two stages: first select the bearings according to the shaft diameter, then check the other design data according to the algorithms presented above, depending on the type of bearing: radial with balls, radial-axial with balls, radial with cylindrical rollers or radial-axial with conical rollers.

10.6. ELECTRIC MOTOR SIZING ALGORITHM

The algorithm uses the deductive strategy to select the electric motor required for the designed mechanism. The user enters the design data on the basis of which the database will be queried: payload, number of gears, gear type, transmission type, data on the inclusion of paired bearings and sliding bearings housings. [88] [89] [65]

Calculate the transmission efficiency, then calculate the required power. The value of the required power will be used to query the Engines table in order to select in the facts base all the engines with rated power > Pnec. A list of information about these engines is displayed: engine type, rated power, speed. The user selects an engine from the list to view and save information about it. The database also allows the verification of the selected motor size, by querying the motor Dimension table in which the specific quotas of electric motors are retained according to their type.

CHAPTER XI

IMPLEMMENTING THE DESIGNED ALGORITHMS

11.1 GENERAL FACTS

The decision support system is made modularly, ensuring high reliability and portability. The database was built in the integrated MySQL Workbench 8.0 development environment, using client-server technology by establishing the connection to the MySQL server.

The designed algorithms were implemented using the integrated development environment Intellij IDEA Ultimate, in the JAVA programming language.

The modular structure of the application offers the possibility to connect the module made in Java to a database with new tables and new information on other components of the mechanisms, by adding new Java modules. Basically, the created structure can be developed very simply by expanding the database and the related Java modules. This is one of the advantages of this application: it is sustainable and can be developed with minimal effort. [23]

Another important advantage of the decision support system is the ability to use databases saved in Cloud, as well as group decision technology, by uploading the application to Cloud. The module made in Java establishes a connection with the MySQL server, no matter where it is located, by using a port and a password. So, connecting to Cloud will be extremely simple, by providing the necessary connection data. In addition, the application can run on multiple VMs to support group decisions. [37]

Another important advantage of the group decision support system is that both development environments used are free for members of the educational community. So any development can be done for free, using the two platforms.

The two modules of the application are:

1) The module made in the JAVA programming language, which includes: the user interface, the implementation of the designed algorithms and the connection with the database;

2) The module made in MySQL, which deals with the creation and management of the database.

11.2. PRODUCT DECSRIPTION

The selection of the calculation module is made from the “Calculator” window which offers the possibility to select the operation desired by the user: “Bearing calculation”, “Engine calculation” or “Bearing selection”.

Following the “Bearing calculation” selection, the decision support system assists the user in sizing a bearing or checking a bearing already selected, according to the design data.

The “Motor calculation” module assists the user in selecting an electric motor according to the design data entered.

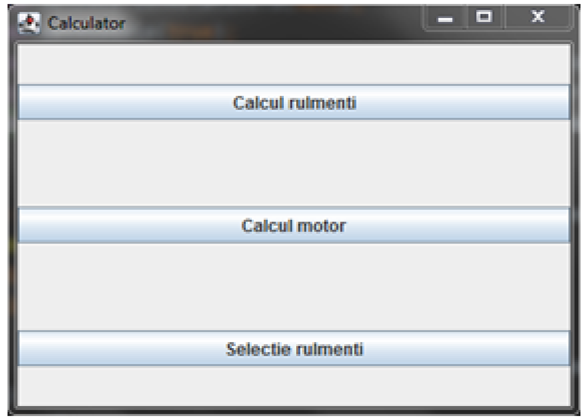


Fig.11.1. Operations select

The “Bearing Selection” module assists the user in selecting bearings according to the price or series of bearing sought by the user.

11.2.2. The “Bearing calculation” module

After selecting the “Bearing calculation” operation, a new window appears in which the user is invited to select one of the operations: “Bearing sizing” or “Bearing check”.
The “Bearing sizing” module is used to design the mechanisms. Through this module the user is assisted in choosing the necessary bearings, based on the design data.

The “Bearing Verification” module will analyze whether a previous choice of a bearing, as a component part of a mechanism, was correct, depending on the design data entered and the series of the chosen bearing.

11.2.2.1. "Bearing sizing" module

The first step in sizing the bearing will be to choose the type of bearing that the user intends to use. In the “Bearing type” window, the user is prompted to choose one of the following bearing types:

- Radial ball bearing
- Radial-axial ball bearing
- Radial bearing with cylindrical rollers
- Radial-axial bearing with tapered rollers

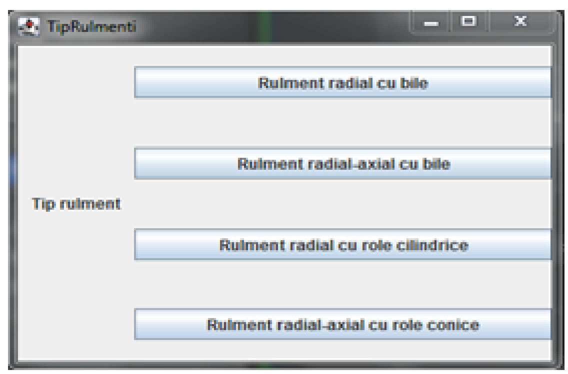
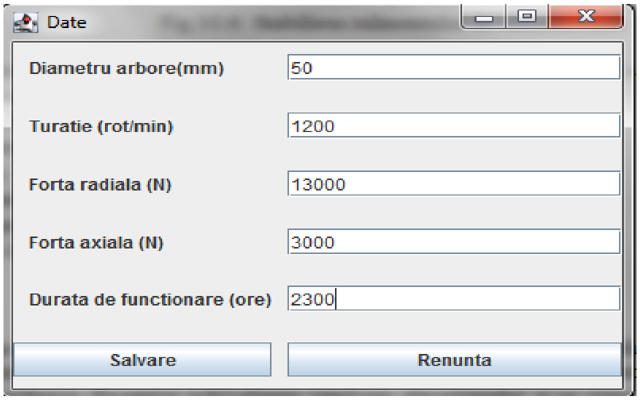


Fig.11.2. Selection of the type of the bearing

Following the selection of the bearing, a new window opens through which the user is invited to choose the type of assembly used and the type of bearing.

The data window that appears after this selection invites the user to enter the design data based on which the bearing will be dimensioned.

- The required design data are:
- Shaft diameter
- Speed
- Radial force
- Axial force
- Allowable operating time.



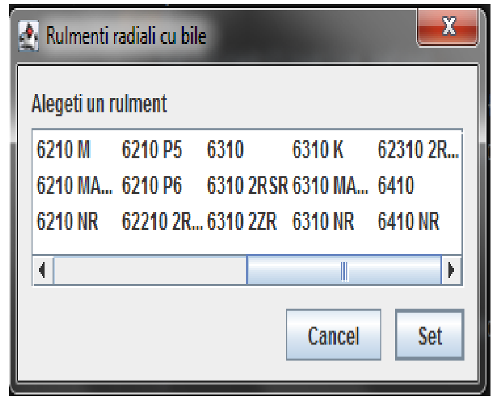


Fig. 11.3 Entering design data

Fig.11.4. List of bearings that can be used

After validating the data entered by the user, the database is queried through the MySQL server and the data is processed according to the calculation algorithm. A list of the series of all bearings that can be used in the mechanism to be designed is displayed.

From the displayed list, the user can select a bearing in order to view all the data related to the chosen bearing: series, name, dynamic capacity, static capacity, speed limit, in parallel with the entered design data.

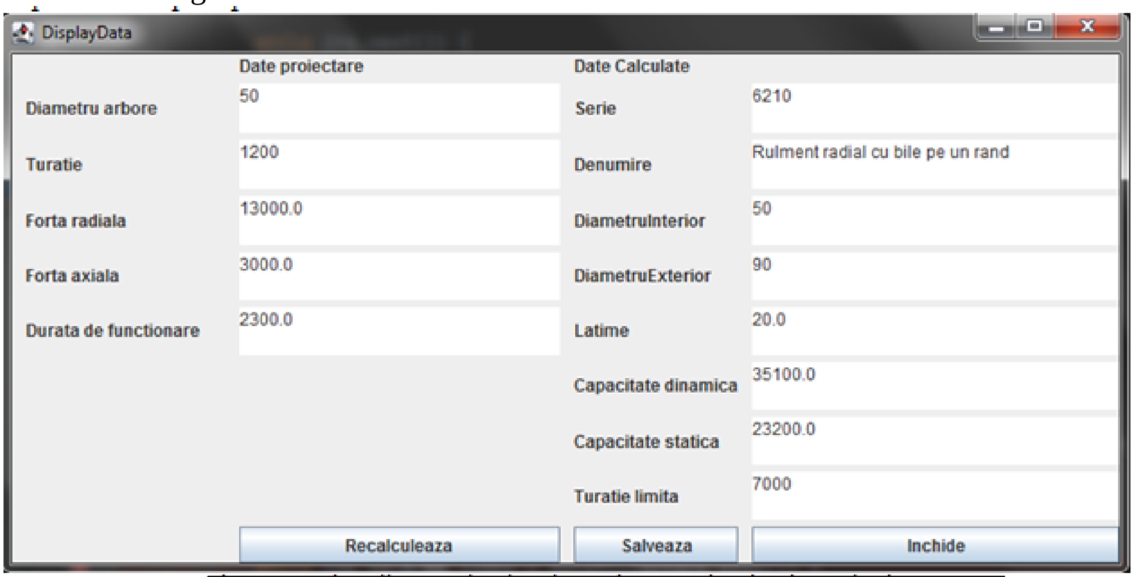


Fig.11.5. Design data and calculated data

The user has the possibility to choose to save the data in a text file in order to print them on a hard copy.

If the chosen bearing does not correspond to the design data, i.e. Lh < Lha then the calculation for a bearing with a larger size is resumed. If this is not a viable construction solution, then the user is instructed to replace the radial ball bearing with a radial-axial roller bearing/tapered roller bearing, which could take on larger loads at approximately the same dimensions.

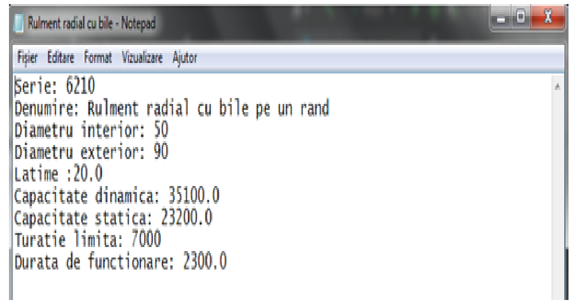


Fig. 11.6. Content of the create file

11.2.2.2. "Bearing check" module

Through this module, the user has the possibility to verify the correctness of the selection of a bearing within a mechanism. In this way, it is possible to calculate and predict possible situations in which the bearings fail due to incorrect calculations or for exceeding the service life.

The user is invited to select the type of bearing to be checked, the assembly data and the design data:

- Bearing series
- Shaft diameter
- Speed
- Radial force
- Axial force
- Permissible service life

Fig.11.7 Data required in order to check a bearing

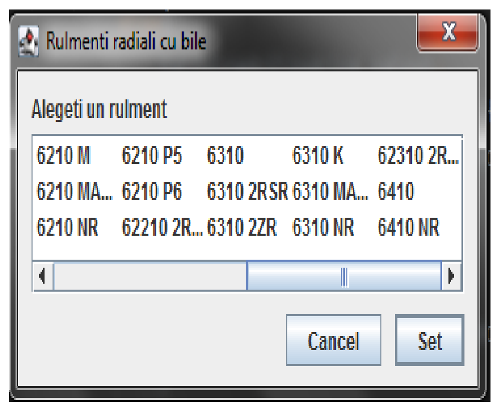


Fig. 11.8. List of bearings that can be used

After entering this data, a list of all bearings that can be used in that mechanism is displayed.

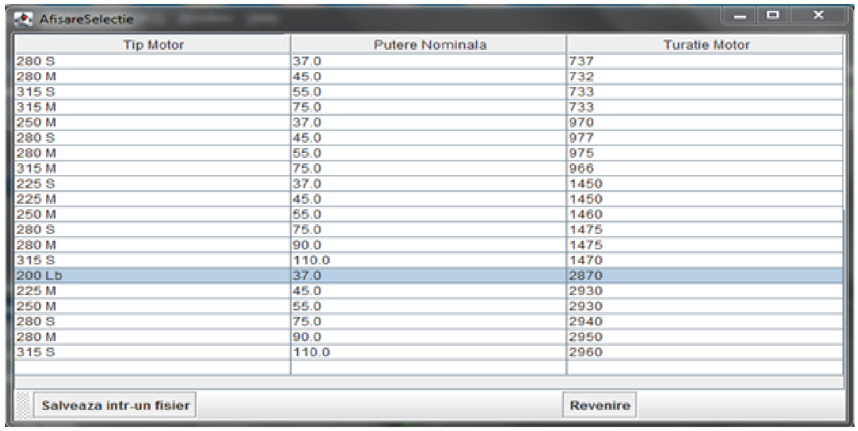
In this way, the user can check if the specified bearing can be used according to the entered design data, but can also view other bearings with which it can be successfully replaced in the designed mechanism.

After selecting the series, the calculated results are displayed in parallel with the entered design data.

The entered data as well as the calculated data can be saved in a file in order to be printed.

11.2.3. "Engine calculation" module

Following the selection of the "Engine calculation" operation, a new window appears in which the user is invited to enter the data necessary for the engine selection: the useful power and the number of steps. Subsequently, the user must specify some data on the designed mechanism: gears with cylindrical wheels / taper wheels, belt driven transmission, pair of bearings, pair of sliding bearings. It is calculated the engine power based on the algorithm shown in Fig. 10.10. The Engines table is accessed via the MYSQL server and a list of all engines that satisfy the Pnominal> Pnecessary relationship is displayed. The list includes data on the respective engines: engine type, rated power, speed. If the user selects “Save to a file”, the results of the database query are saved in a text file, which can be printed later.



Tip Motor	Putere Nominala	Turatie Motor
280 S	37.0	737
280 M	45.0	732
315 S	55.0	733
315 M	75.0	733
250 M	37.0	970
280 S	45.0	977
280 M	55.0	975
315 M	75.0	966
225 S	37.0	1450
225 M	45.0	1450
250 M	55.0	1460
280 S	75.0	1475
280 M	90.0	1475
315 S	110.0	1470
200 Lb	37.0	2870
225 M	45.0	2930
250 M	55.0	2930
280 S	75.0	2940
280 M	90.0	2950
315 S	110.0	2960

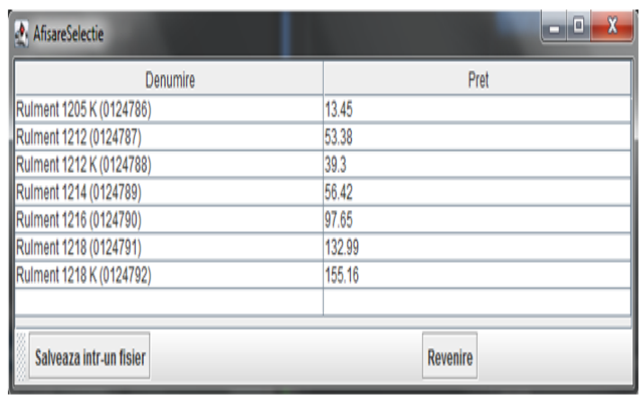
Fig. 11.10. List of the motors that can be used for the designed mechanism

11.2.4. “Bearing Selection” Module

Through this module the user has the possibility to query the database containing the bearings, according to certain criteria, namely: price or series.

If the user selects “Selection by price”, then he will be prompted to enter the maximum value of the price of the bearings he wants to view. After entering this value, the database tables are queried via the MySQL server and all the bearings that meet the entered criterion are retained in an intermediate structure.

A list of selected bearings is then displayed, with each bearing in the list showing the series and price. The displayed list can be saved in a text file in order to be printed later.



Denumire	Pret
Rulment 1205 K (0124786)	13.45
Rulment 1212 (0124787)	53.38
Rulment 1212 K (0124788)	39.3
Rulment 1214 (0124789)	56.42
Rulment 1216 (0124790)	97.65
Rulment 1218 (0124791)	132.99
Rulment 1218 K (0124792)	155.16

Fig. 11.11. Display of results of bearings selection by series

If the user selects “Selection by series”, then he will be prompted to enter a few characters from the series of bearings he wants to view. After entering these values, the database tables are queried via the MySQL server and all the bearings that meet the entered criterion are retained in an intermediate structure.

A list of selected bearings is then displayed, with each bearing in the list showing the series and price. The displayed list can be saved in a text file in order to be printed later.

CHAPTER XII

EXPERIMENTAL RESULTS

The algorithms created and implemented through the Java programming language and the facilities offered by MySQL, were used in sizing or checking bearings, respectively, in the selection of electric motors.

Through these concrete cases, the functions and calculation modules of the support system for group decisions in the design of mechanisms were verified, the results being identical to those obtained in the literature [2], [12], [57]. At the same time, the correlations of numerical data with specific knowledge were verified.

The algorithms created and implemented through the Java programming language and the facilities offered by MySQL, were used in sizing or checking bearings, respectively, in the selection of electric motors.

Through these concrete cases, the functions and calculation modules of the support system for group decisions in the design of mechanisms were verified, the results being identical to those obtained in the literature [2], [12], [57]. At the same time, the correlations of numerical data with specific knowledge were verified

12.1. APPLICATION 1

Consider a radial ball bearing on a row mounted in the gearbox of a vehicle. [51] In the most disadvantageous conditions, this bearing will work as follows:

Axial force = 4760 N
Radial force = 14320 N
Shaft diameter = 60 mm
Speed = 400 rpm

A bearing must be dimensioned to operate for at least 6000 hours under these conditions.

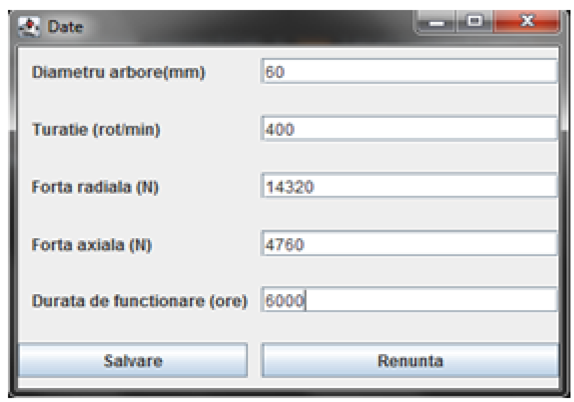


Fig. 12.1. Design data introduction

The mechanism has an assembly in which the specified bearing will be conductive. Following the database query, a list of bearings is obtained, including the 6312 series bearing which is used in the gearbox of Volvo, Scania cars etc. [3].

Bearing 6312 has dynamic capacity C = 81900 N and static capacity Co = 52200 N. [12]
We calculate the Fa / Co ratio = 4.76 / 52.2 = 0.091

From the FaCo table we obtain e = 0.28 and y = 1.55.

We calculate the Fa / Fr ratio = 4.76 / 14.32 = 0.33. It is observed that Fa / Fr> e, so the dynamic load will be calculated with the relation (10.1), where x = 0.56 and y = 1.55.

Therefore,

$$P = 0.56*14320 + 1.55 * 4760 = 15397 \text{ N} \quad (12.1)$$

Calculate the service life with the formula (10.2):

$$L= (81900/15397)^3 = 150.671 \text{ mil rot} \quad (12.2)$$

The operating time in hours is calculated with the formula: (10.3):

$$Lh = 10^6 * 150.671 / (400*60)= 6277 \text{ ore} \quad (12.3)$$

As Lh> admissible operating time of 6000 hours, the 6312 series bearing can be used on the designed gearbox.

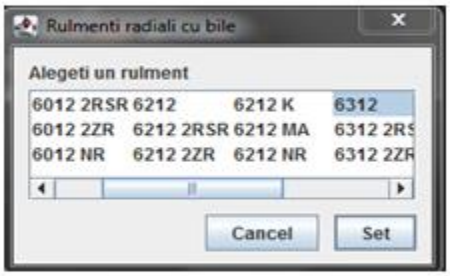


Fig. 12.2. List of bearing with balls on one line

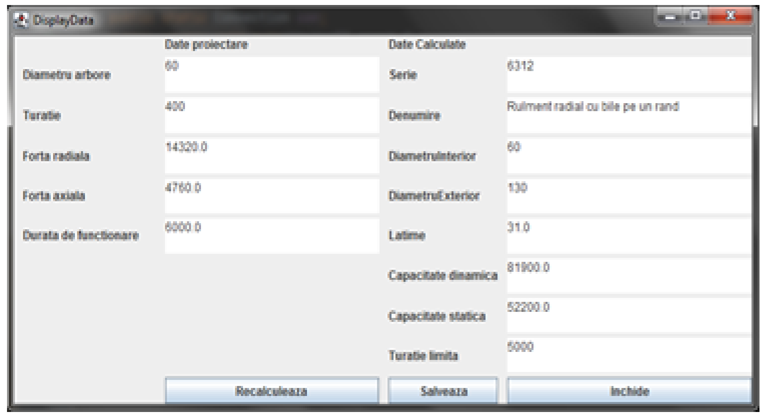


Fig. 12.3. Display of calculated data

12.2. APPLICATION 2

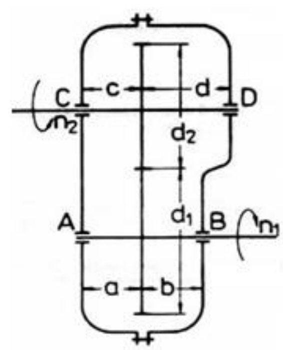


Fig.12.4 kinematic diagram of the reversing gearbox

It is considered the following inverter reducer. The necessary bearings in points A, B, C, D must be dimensioned.

The following design data are known: [12]

Input power = 97 kW

Input shaft speed n1 = 1000 rpm

Axial force Fa = 4000 N

Operating time Lha = 10000 hours

Dimensions:

a = b = c = 38 mm

d = 52 mm

d1 = 172 mm

d2 = 148 mm

Since a = b = c = 38 mm, bearings with a diameter of 35 mm will be selected.

It is calculated the output shaft speed: [12]

$$n_2 = \frac{n_1 * d_1}{d_2} = \frac{1000 * 172}{148} = 1162 \; rot/min \quad (12.4)$$

It is calculated the time of entry:[12]

$$M = \frac{9.74*97}{1000} = 0.94 \; kNm \quad (12.5)$$

The tangential forces on the wheels are calculated 1 and 2:[12]

$$T = \frac{M}{0.5 * d_1} = 10.93\ kN \qquad (12.6)$$

It is calculated the repulsive force between the 1st and 2nd wheels:[12]

$$R = T * tg\ \alpha = 10.93 * tg\ 20^0 = 9.07\ kN \qquad (12.7)$$

It is calculated the load force of shafts 1 and 2 [12]

$$F = \sqrt{T^2 + R^2} = 11.63\ kN \qquad (12.8)$$

It is calculated the bearing load forces: [12]

FrA=FrB = F/2 = 5.82 kN (12.9)

FrC= F*d/ (c+d) = 6.72 kN (12.10)

FrD= F* c/(c+d) =4.91 kN (12.11)

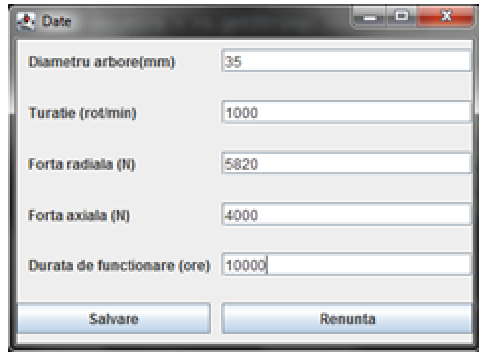


Fig. 12.7. Design data for points A and B

Radial cylindrical roller bearings are chosen for all points.

The dynamic loads required for the bearings are calculated at each point: [12]

$$C_A = C_B = 5.82 * \sqrt[\frac{10}{3}]{\frac{60*10000*1000}{1000000}} = 39.66\ kN \qquad (12.12)$$

$$Cc = 6.72 * \sqrt[\frac{10}{3}]{\frac{60*10000*1162}{1000000}} = 47.91\ kN \qquad (12.13)$$

$$C_D = 4.91 * \sqrt[\frac{10}{3}]{\frac{60*10000*1162}{1000000}} = 35\ kN \qquad (12.14)$$

Se va utiliza rulmentul NJ207E întrucât acest rulment satisfice restricțiile tuturor punctelor A,B,C și D ale reductorului inversor din figură. The NJ207E bearing will be used as this bearing satisfies the restrictions of all points A, B, C and D of the reversing gearbox in the figure.

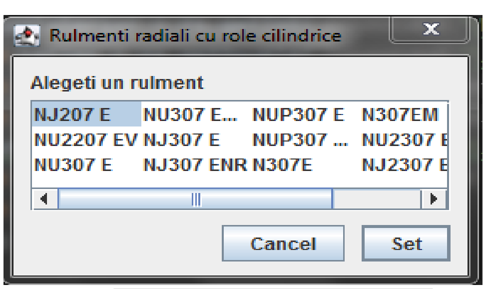


Fig.12.8. The results

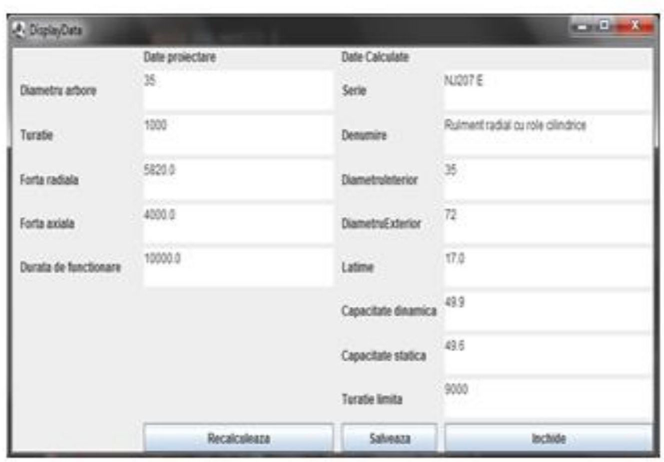


Fig. 12.9 Data display

CONCLUSIONS

C.1. GENERALCONCLUSIONS

The group decision support systems are an expanding field of IT applications. With the development of communication technologies, Informatics, programming languages, new research directions will be created in this field.

The purpose of the group decision support system is to support managers at various hierarchical levels or design engineers in solving selection problems, and/or designing optimal solutions, which solve the various problems they face in their daily work. The support system for group decisions can be seen as a tool and not as a substitute for the human factor in the decision-making process.

Given the increasing complexity of technical solutions, but also the limited resources that any organization has at its disposal, the decision becomes a key element. Making good decisions involves using a large amount of data and information that cannot be managed without a decision support system.

The support system for group decisions provides communication channels, the environment in which specialists can collaborate, the tool through which managers coordinate work teams in various parts of the world. The design of a decision support system must take into account the scope, size of the organization, the number of users forming the various working groups, the amount of data expected to be stored, updated, selected, processed and transmitted via the Internet or other communication channels.

In the current context of computerization, a decisive role is played by intelligent computer systems, which are based on reasoning with knowledge-based information, as opposed to traditional, data-based ones.

The necessity and utility of this doctoral thesis is highlighted by the following arguments: the rarity of support systems for dedicated group decisions in the field of mechanical engineering and the non-existence of support systems in the design of mechanisms based on Romanian products; the varied but incomplete range of software products that can be used for mechanism design and design activities; the very large volume of data and specialized information which causes a great difficulty in their management by experts in the fields of design and design of mechanical products.

In order to deal with the proposed topic and objectives, a bibliographic study was carried out and structured algorithms were designed to allow the development of programs integrated in a decision support system containing the database and inference modules.

By verifying the functions and calculation modules through factual cases, the flexibility and efficiency of the designed and implemented support system was demonstrated. In addition, the level of training of the system user may be lower than that of a user of a classic system, which results in greater addressability to a wider range of users.

C.2. ORIGINAL CONTRIBUTIONS

In order to design some algorithms for the support system for group decisions dedicated to the design of the mechanisms, the design algorithms for the design of these systems were analyzed, as well as the types of data structures that can be used in this field.

From the multitude of equipment, devices and subassemblies components of the mechanisms, two categories were chosen: directional bearings and electric motors.

Considering the studied bibliography, the developed support system represents the first step in creating an application that consists of databases containing local products and based on theoretical and practical knowledge of specialists in the field of mechanism design.

The paper highlights from the beginning, based on bibliographic research, general and specific objectives as well as the adapted research strategy. In addition, the paper highlights and justifies the models, algorithms and logic schemes designed and implemented through edifying results as well as through user-friendly interfaces.

Through a close collaboration with experts in the field, sets of knowledge necessary for the design of mechanisms using bearings were identified, respectively sets of knowledge necessary for the design of drive systems of robots using direct current motors. In order to select a way of representation that is as efficient as possible and that offers the possibility to solve a wide range of problems, the ways of representing the knowledge of an expert system were analyzed. Following the collection of all this information, the structure of the database was designed so that it stores the knowledge received from human experts in the field. In designing the structures of the tables that are contained in these databases, the aim was:

- Ensuring the shortest response time of the support system. This goal was achieved due to the design of the tables according to the classes of problems to which the system will respond. Thus, for bearings, the assembly-bearing type combination methods were treated separately, the database queries on this field obtaining a direct answer from the tables dedicated to the subject.
- Establishing relationships between the tables of the database leading to the immediate determination of the rules that can be applied in various cases of combining the premises.

These steps were performed both for the modules that deal with sizing or checking the bearings, and for the module that designs the direct current motors. A database has been developed that includes material characteristics and construction parameters according to the standards.

The database was developed through the MySQL package and can be easily accessed by the support system modules and, on the other hand, can be used separately. Through specific techniques, the database tables were populated with bearings produced in Romania.

Through the MySQL server and the multiple connection possibilities offered by it, the database can be used in cloud services, internet and/or through connections to various modules made in Python, Java, NET framework, Node.js, and so on. Downloading and installing the platform is free, so using it together with the MySQL Workbench module that ensures a simple, intuitive view of all the information in the database tables, offers any user, in any location, the possibility to connect to the created support system. In addition, for each table in the database there is a form that ensures the entry of data in a very easy way for a user that has no specialized knowledge.

It was performed a critical analysis of the expert systems and of the support systems for decisions from various fields, highlighting, especially, the evolution of those in the field of industrial engineering. This analysis compares in a multicriterial way the data modeling variants in various fields as knowledge, and the structures of expert systems, respectively of the developed support systems, based on inference algorithms (reasoning).

In close collaboration with the specialists in the field, it was analyzed the reasoning by which they design the mechanisms that include the two categories of selected components. Since the chosen way of representing knowledge is based on production rules and frameworks, the determined production rules were implemented through the relationships between the fields of the knowledge base tables.

The inference mechanism is based on the algorithms for solving the design problems of the bearings, respectively of the direct current motors. Algorithms use complex data structures to store input and intermediate data. In addition, the algorithms can also work on dynamically allocated data structures, graphs or trees, for which case search algorithms are used by traversing graphs in width or depth or algorithms for determining the minimum cost path, respectively minimum cost tree construction algorithms. This last category of algorithms can also offer solutions in the case of designing mechanisms with many components having a selection criterion that optimizes the solutions based on maximum/minimum values. The search algorithms have been implemented and are listed in Annex 1 and Annex 2, respectively.

From the analysis of the knowledge taken from human experts, the sizing and verification algorithms of the bearings were built, as well as the selection algorithm of the electric motor. Based on them, the classes, functions and input/output parameters necessary for the implementation of the respective algorithms were created.

The general algorithm is divided into distinct modules (sizing, verification, selection, etc.) to which expert subsystems are associated with associated program modules (assembly type, bearing type, etc.).

Original modules have been designed for the design of various components of the mechanisms.

The implementation of the algorithms was done in the Java language, through the Intellij IDEA platform. This object-oriented programming language provides the ability to run programs on any Java Virtual Machine (JVM). Thus, the program can run on mobile phones, workstations or server-client systems, with minimal modifications.

The developed modules have independent structures that can be operated individually but also through the general algorithm. Thus, users are allowed to make several variants at the level of subassemblies, under the conditions of detailed information by the system, in order to obtain optimal solutions. For each module developed at detail level, graphical user interfaces have been designed for data entry, for monitoring their processing as well as for post-processing the results.

The case studies for design, sizing, verification, clearly highlight the functionality of the developed decision support system, by going step by step through five applications starting from the data entry to the generation of results.

The solutions obtained by the support system can be communicated in a very easy way using the local computer network or the internet. The data structures used to represent the solutions do not require a large memory space or long transmission time. In addition, the portability of these data structures across multiple environments or programming platforms is ensured, leading to a widespread use of the results.

In order to design the support system for group decisions, the ways of accomplishing the group decision-making process, the necessary components of this system and the decision situations that may arise were analyzed. Particular attention was paid to avoiding influencing the decisions of some participants as a result of which unique, “group” decisions can be formed.

The decision support system allows users to select at each step, from the multitude of solutions found by applying production rules, those with which the user wants to continue the process. This process can be performed by each decision maker, and the results obtained by each manager or design engineer in the decision maker group can be memorized for further negotiations. All decision makers start with the same input data, but can get different solutions depending on the selections they make at each step.

Finally, the group of decision makers analyzes the solutions discovered by each member of the group and negotiates a final solution. In this way, the support system for group decisions does not allow forming unique decisions by influencing some members of the group.

In addition, in the negotiation, each decision-maker can come up with his own system of explanations of the steps taken at each step. Through debate, the group of decision makers can follow the reasoning of each member and can build together a new solution, based on the experience of the whole group, assembling the solution from the component parts of the solutions presented by the group members.

The interface of the group decision support system is very friendly, any user can use the application because each step is based on questions and answers.

The results obtained demonstrate the advantages of such a decision support system for engineering applications, in particular for sizing and checking bearings and for the selection of electric motors. Articles presented at various scientific conferences and published in journals have already been cited 17 times according to [6].

The paper presents all stages of the development of a support system for group decisions paving the way for the generalization of this technique for all design applications in the field of industrial engineering.

C.3. PERSPECTIVES FOR FURTHER DEVELOPMENT

The group decision support system allows for a wide range of developments. First of all, this is possible due to the portability of the algorithms created. They work in any programming language and can be used on dynamically allocated data structures, which will make the allocated memory space more efficient.

By implementing an algorithm that ensures the operation in the client-server system, all stored data can be shared by members of the decision group and synchronous working meetings can be organized, regardless of the location of the decision makers. In this regard, it is extremely important to ensure a developed security system for both the data in the database and the users participating in the negotiation meetings.

The group decision support system can be developed by adding new components of the expert system to support the design, by storing and processing knowledge about other subassemblies of the mechanisms.

The group decision support system is a step towards the computerization of a difficult area, namely design. The created algorithms support the design engineers and managers in the design process and the decision on an optimal solution according to the imposed criteria. In addition, through the explanation module, the included expert system supports students in creating correct reasoning in the field of mechanisms design.

The support system for group decisions in the design of mechanisms is a tool made available to specialists in order to determine the solutions of design problems in an efficient and correct way.

SELECTIVE BIBLIOGRAPHY

2. *** “Catalog reductoare coaxiale”, http://www.reductor-motor.com/rom-redcoax.html
10. *** http://iota.ee.tuiasi.ro/
12. *** http://www.urb-s.ro/ro/products.html
21. *Bizoi M*, “Referat 1”, http://www.referat_1_Bizoi_web.pdf
27. *Butilă E.V.*, “Rezumat teza de doctorat”, Universitatea Transilvania din Brașov, 2009
32. *Carstoiu, D.I.* „Sisteme expert”, Editura ALL, București, 1994, pag 16-26
48. *Filip F. G.,* „Decision support and control for large-scale complex systems. Annual Reviews in Control*”,* ,2008
49. *Florea R.,* “Organe de maşini. Reductoare”, Editura Universităţii „Lucian Blaga” din Sibiu, 1997
52. Gafiţanu M., ş.a., “Organe de maşini”, vol. Işi II. Editura Tehnică, Bucureşti, 1981
57. *Geonea I.D.,* „Calculul și proiectarea reductoarelor”, Universitatea din Craiova, 2018
65. *Ianuș G.,* “Transmisie mecanică cu redactor și curele trapezoidale înguste”, Editura Politehnium, Iași, 2010
79. ***Moşneaga A.O.,*** *Tempea I.,* “Folosirea sistemelor expert pentru proiectarea mecanismelor și a sistemelor de acționare electrică”, CONFERENG 2018, Târgu Jiu, 2018
80. ***Moşneaga A.O.,*** *Tempea I.,* “Utilizarea bazelor de date pentru proiectarea motoarelor de curent continuu destinate acționării mecanismelor”, CONFERENG, Târgu Jiu, 2018
81. ***Moşneaga A.O.,*** *Tempea I.,* „Folosirea sistemelor expert pentru proiectarea mecanismelor”, SYMECH 2018, Târgu Jiu, 2018
82. ***Moşneaga A.O.,*** *Tempea I.,* „Utilizarea bazelor de date pentru proiectarea rulmenților din construcția mecanismelor”, SYMECH 2018, Târgu Jiu, 2018
83. ***Moşneaga A.O,*** „Arhitectura sistemelor expert utile în monitorizarea mecanismelor roboților”, Referat nr.1 din cadrul tezei de doctorat, Universitatea Politehnică Bucuresti, 2017
85. Mușat M, Stoica G., “Transmisii mecanice cu reductoare într-o treaptă”, 2004
88. Palade V., “Reductor de turație cu o treaptă”, Universitatea “Dunărea de jos”, Galați, 2008
103. *Zaharie D., Năstase P.,* “ Sisteme expert de gestiune”, Editura Romcart, București, 1993