Use of Fuller-Polya diagram for teaching engineering problem solving in undergraduate design classes

Gustavo J. Molina

School of Technology, Georgia Southern University

Abstract

On teaching a sophomore Engineering design class the author emphasizes a problem solving approach to teaching which integrates through team-work design miniprojects. The three levels of design-problem complexity, e.g., routine standard, poorly defined and open-ended creative-design problems, are introduced in a studio-lab classroom setting. The last two problem types are readily solved by sophomores after they learn basic techniques. However, the routine single-answer standard-algorithms may be difficult for students if not presented as straight "plug-data" computations (i.e., if formulas or procedures are not for explicit computation of the required results from the given data). Sophomores are not used to make the connections between the mathematics and physics they learned and the standardized handbook-type engineering computations, where multiple data and result values may be required or produced, and non-algebraic procedures are employed.

The author introduced in his class the Fuller-Polya diagram for problem solving, a simple structured method outlined by Fuller and Polya and formalized by Kardos, to graphically organize the variables and their relationships in the computation without the mathematical formula and procedure details. The approach showed useful to help students' understanding and insight of deterministic engineering algorithms.

Introduction

Teaching students how to solve problems is a growing concern of Engineering and Technology education. Problem solving in the Engineering/Technology practice is usually directed to the design of products or processes, and this connection makes the teaching of engineering problem solving a natural part of design classes. In recent years new undergraduate courses have been implemented that expand creative thinking in engineering design by including solving problem and project based-learning [1]. A number of innovative teaching techniques can be employed for such purpose, being the "studio" methods particularly successful to enhance student creativity and involvement in solving problems, mainly through team-work under a mentor [2].

A problem-based teaching methodology is used by the author in the Creative Decisions and Design sophomore class he teaches as part of the Georgia Tech Regional Engineering Program (GTREP) and Regents Engineering Transfer Program (RETP) for Mechanical Engineering majors in Georgia Southern University. The different problem-solving techniques integrate in team-work design projects. The class meets twice a week for 50-minute lectures, and once a week for three-hour studio-lab in an appropriate classroom for hands-on work, as well in special labs for some activities (i.e., machining lab, robotics and measurement lab, etc.)

The three levels of design-problem complexity, e.g., routine standard algorithms, poorly defined problems and open-ended creative-design problems, are introduced in the lectures and applied in the studio-lab classroom. The last two problem types are readily solved by sophomores after they learn basic techniques (e.g., brainstorming, QFD, generation of alternatives, morphological method, TRIZ, etc., see reference [3]). Surprisingly, routine standard engineering algorithms may sometimes be difficult for students if not presented as straight "plug-data" computations (i.e., if formulas or procedures are not straightforward explicit computation of the required results from the given data). Most of the standard components in detailed mechanical engineering design are defined or chosen by a combination of computations and/or selection methods. These computations or methods can be found in manuals, catalogues, standards and upper-level textbooks but they are not directed to the non-experienced sophomore.

Engineering students lack of algorithm insight and manipulation ability has apparently not been investigated. One reason for such difficulties may be that students have to apply in the context of the design process the knowledge that they acquired in strictly formal mathematics classes: Klebanoff and Winkell [4] noted the compartmentalization that exists in engineering programs, in which students see little substantive relationship between math, science and engineering. They speculated that the type of symbolic manipulations that students are asked to perform in mathematics classes does not prepare them for applying mathematical concepts in engineering contexts. The author and colleagues [5] systematically investigated if the disconnect between mathematics teaching with x and y as preferred variables and the use of more varied and descriptive names in engineering and technology courses can be an explanation for students' difficulty in solving mathematically simple problems in engineering applications.

Sophomores seem not used to make the connections between the mathematics and physics they learned and standardized handbook-type engineering computations, where multiple data and result values may be required or produced, and non-algebraic procedures are employed. For the student with no experience in design the understanding of such handbook-type engineering algorithms would be limited to the data at hand, from which a solution is obtained mainly through algebraic manipulation and without insight into the variables' relationships. These general observations prompted the introduction of the Fuller-Polya diagram (or map) in the author's class.

Diagrams and maps play a key role in engineering problem solving, and they are essential tools for teaching problem-solving. Diagrams naturally appear in problems involving physics because such problems require the solution of geometric subproblems, but also because graphs serve as formal communication in science and engineering [6]. Humans often use diagrams when solving physics problems: Larkin and Simon [7] described the psychological and computational advantages of using diagrams in problem solving, among them (i) diagrams focus attention on elements relationships and they reduce search because related elements are usually close together, (ii) diagrams minimize unwanted information, (iii) they facilitate perceptual inferences and recognition of problem-solving methods that may be applicable, and (iv) diagrams allow quick procedure checks.

The Fuller-Polya diagram

The Fuller-Polya diagram (FPD) for problem solving is a simple structured method outlined by Fuller [8] from a Polya's [9] suggestion and further formalized by Kardos [10]. It graphically organizes the variables and their relationships in the computation, including non-algebraic procedures, without the mathematics formula and procedure details.

The FPD is indicated for deterministic problems for which a solution can be produced from the available data by using explicit relationships, provided that data is sufficient and not contradictory. It is not useful, however, for open-end problems as the typical creative-design and generation of alternatives problems, for which requirements may be poorly defined and/or contradictory. The FPD shows the structure of the solution independent of the computations and procedures, but focusing attention on the existence of relationships between variables rather on any fixed "recipe" to produce an answer.

The FPD methodology requires the following definitions and standard symbols:

Variable is a known or unknown "value" (it may be a number, an interval or even a code that identifies a standard part). The symbol for variable is a circle enclosing the variable name as presented in Figure 1.

Reversible algorithm is a computation or a sequence of computations that can be carried out in any "direction", even if algebra or mathematics manipulation may be needed to "reverse" such direction. The symbol for reversible algorithm is a square with an order number inside as presented in Figure 1.

Irreversible algorithm (or meta-operation) is a procedure that must be carried out only in a given "direction" (i.e., a choice between a set of alternatives is an irreversible algorithm). The symbol for irreversible algorithm is a diamond with an order number inside as presented in Figure 1.

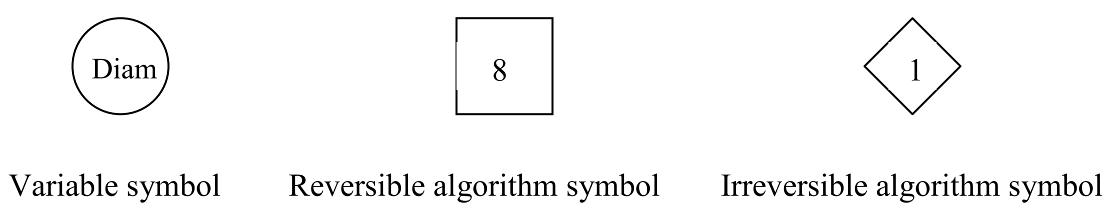

Figure 1. Standard symbols for Fuller-Polya diagram

Flowlines connect each variable symbol to every algorithm symbol where the variable value is used in or produced from. Several examples of FPDs were presented in the work of Kardos [8]. Figure 2 presents a simple FPD (as developed by one student in the author's class) for the diameter design of a pump solid-shaft under the constraints of maximum torsional stress and critical speed to reach the first natural frequency for shaft deflection; the calculation procedure can be found in a standard computation handbook [11] and a summary is included in this paper Appendix.

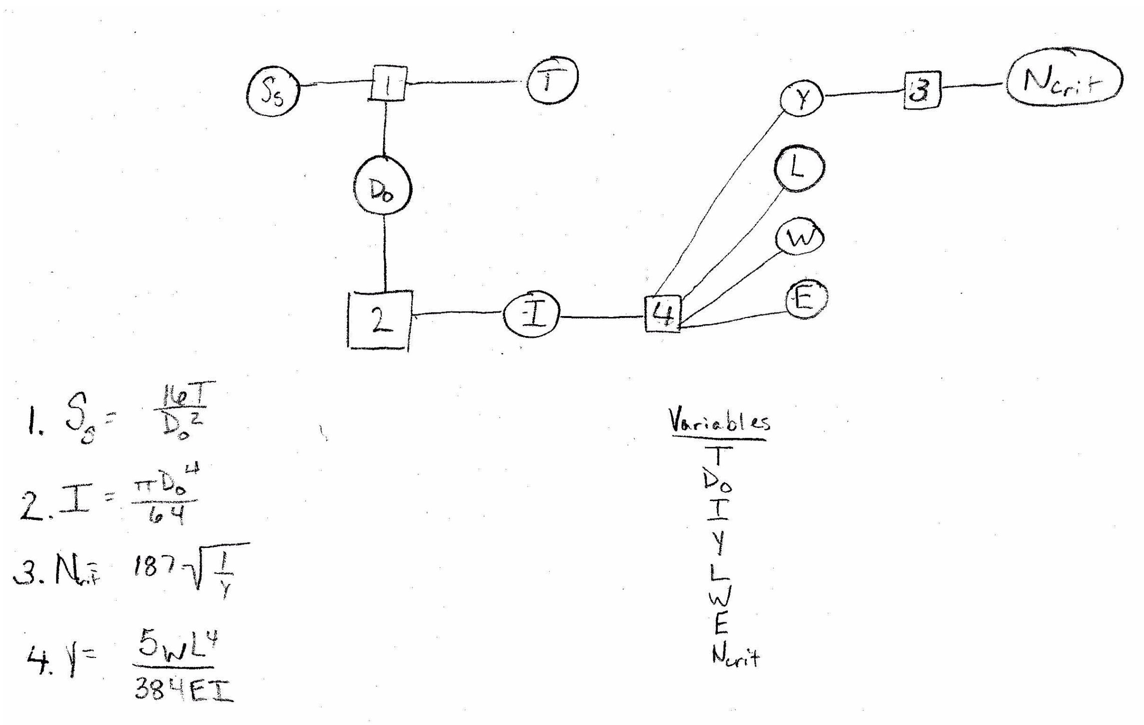

Figure 2. Fuller-Polya Diagram for diameter computation of pump solid-shaft
(See Appendix for summary of computations from reference [11]).

A consecutive numbering for the set (or list) of reversible and irreversible algorithms relate to corresponding algorithm descriptions that are listed on a side. It is not needed to know the exact form of the algorithms to construct a FPD, but it is enough to know that such relationships should exist (i.e., from physical reasoning) or that they can be established (i.e., by measurement and/or experiment). A list of variables names (and descriptions) may be included.

The suggested methodology to construct a FPD is based on Polya's four-step method for problem solving [9]. In the context of FPD these four steps are: (i) the problem must be understood in its physical and geometrical meaning, (ii) the data (either input or output) must be identified, but not necessarily its values, (iii) the relationships between variables must be identified, but not necessarily be known in detail (it is enough to know that a relationship must exist), and (iv) symbols are drawn according to (ii) and connected by flowlines according to (iii); if all of the above is not entirely correct at the beginning, the construction should help clarifying the problem. The FPD methodology may be a complement of problem-solving methods as the Polya's work and its further development by Wales et al. and their GENI idea [12]. While these heuristics methods help planning a mathematical reasoning from the goal (e.g., the unknown) to the equations for solving, the FPD proposes a formal language and graphics approach when the algorithms (i.e., equations or procedures) are known to exist.

The method can be very helpful for identifying the relations between "variables" (e.g., dimensions, parameters, factors, etc.) and "computations", and it is general enough to show if a path to a solution can or cannot be produced from given data, even if the relationships (algorithms) are not yet fully available. However, applications of the FPD methodology are very scarce in the available literature: Vidal and Becker [13] applied the Fuller-Polya map to successfully illustrate in a simple diagram the rather complicated design optimization of a coolant distribution system solution using Lagrange multipliers.

Examples of student applications of the FPD

The FPD methodology was introduced to the authors' Creative Decisions and Design class in the spring semester of the year 2004. The methodology was briefly lectured to the class, after which each student was handed a different standard mechanical-component computation as presented in either a computation handbook [11] or an advanced machine-component-design textbook, as Shigley et al.'s [14]. Students were then given half-hour to develop the corresponding FPDs; examples of student work are presented in Figure 2 and following figures 3 to 5. The main purpose of this exercise was exploring the applicability of the method at sophomore level.

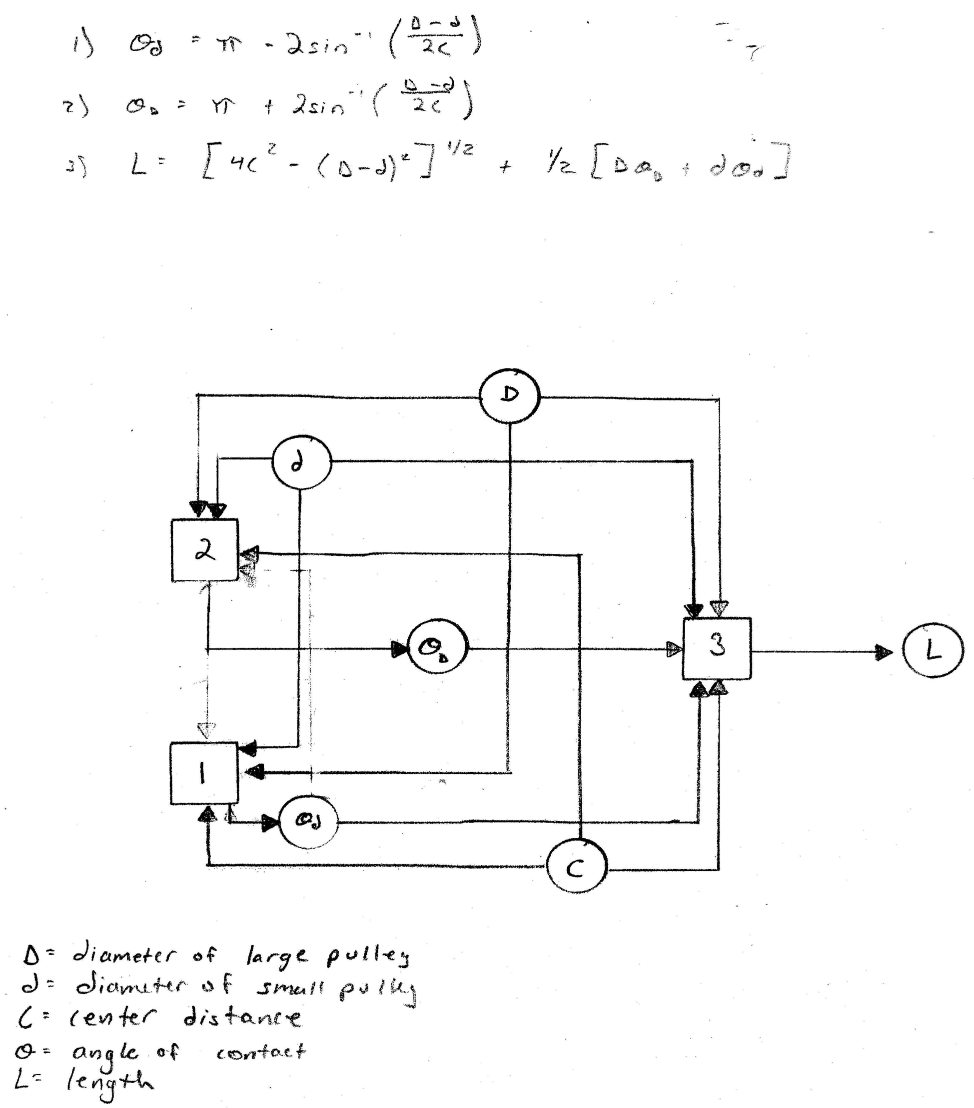

Figure 3. FPD for belt-drive computation (See Appendix for summary of computations from reference [13]).

Figure 4 shows a rather complex problem that is well above sophomore background and experience; although the FPD is not complete, the application of the method allowed the student identifying all variables and algorithms, and most relationships are displayed. The difficulty of the assigned computations ranged from the relatively simple (three to four algorithm) computations of figures 2 and 3, to the more complex corresponding to figures 4 and 5.

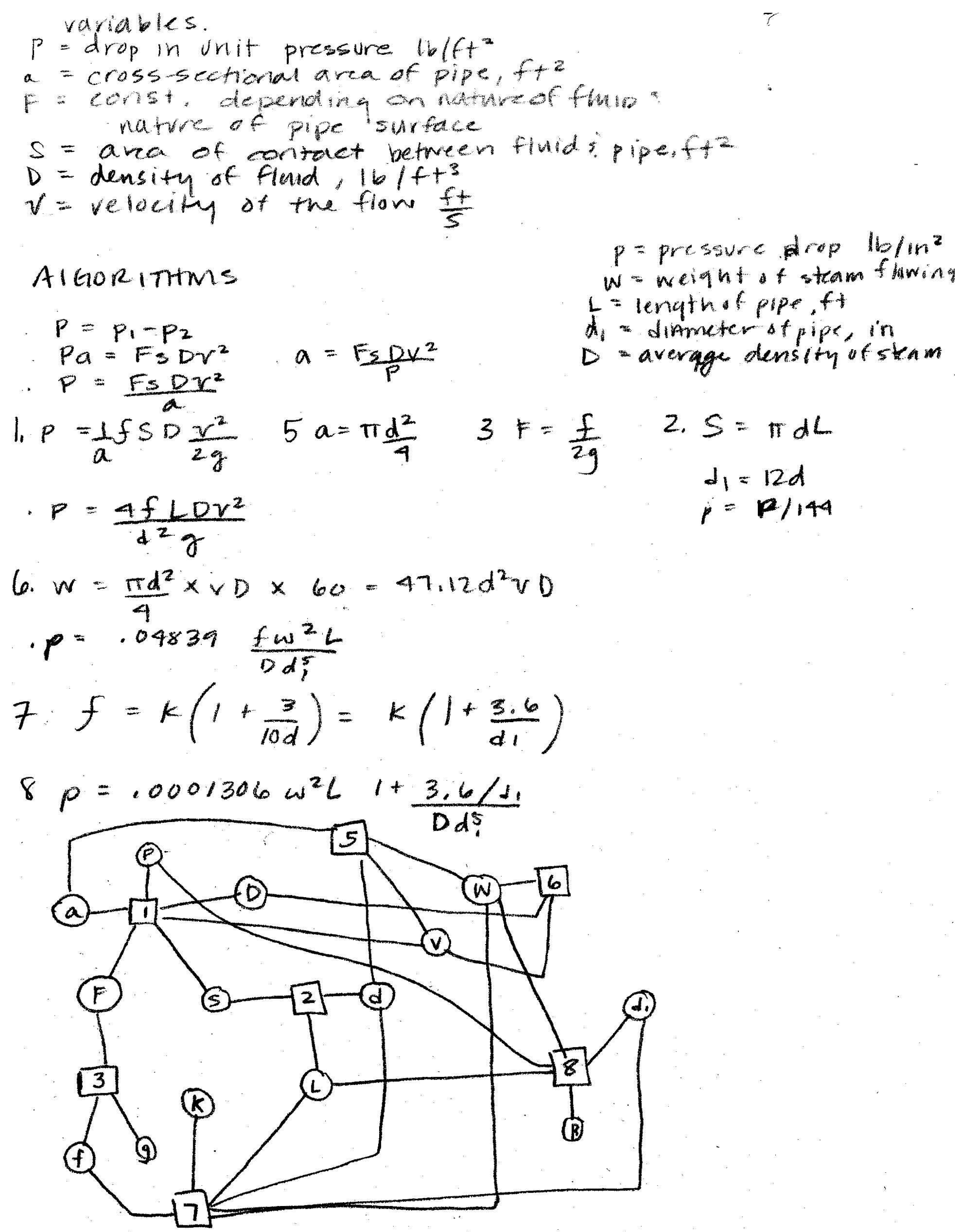

Figure 4. FPD for the computation of pressure-loss in steam piping, see reference [11].

The author is planning a more formal assessment of the technique in his present and future sophomore design classes: the student success and difficulties when dealing with this paper's exploratory exercises are used to design such assessment.

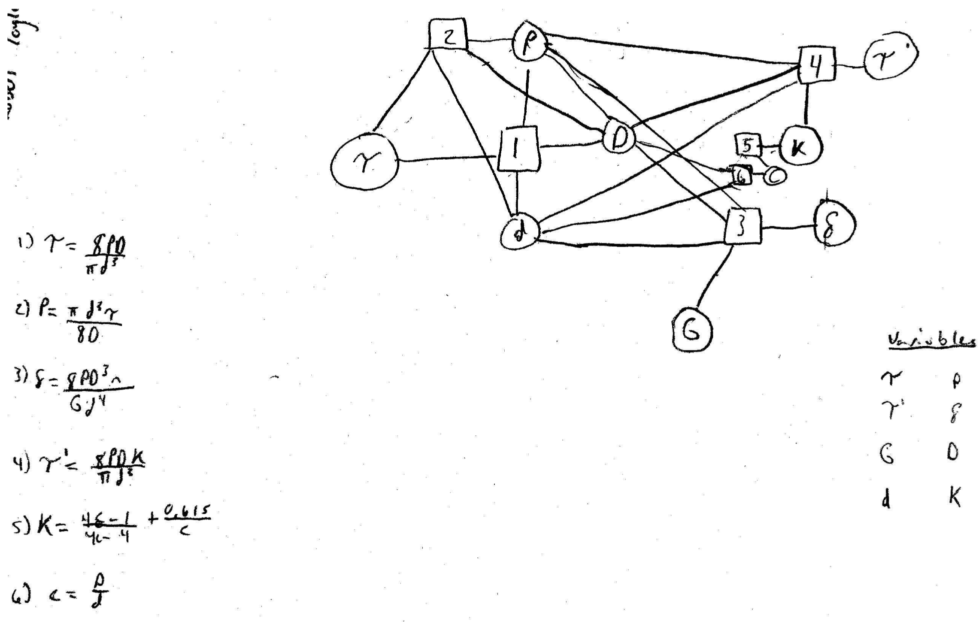

Figure 5. FPD for the computation of helical compression or tension spring, see reference [11].

Conclusions

The main results of this exploratory study are (i) showing the feasibility of introducing the FPD at the level of a sophomore design class and (ii) exploring the students performances when dealing with exercises of varied difficulty using the technique; these results can help the design of a formal assessment of the FPD technique. The introduction of the FPD methodology in the author's design class seemed useful for guiding student analysis and understanding of the problem at hand, because the methodology prompts to fully identify variables and relationships (e.g., algorithms) and to "see the whole picture" before attempting any solution.

The method also may help the student insight of engineering calculations, because it shows, for instance, that variables may be either given data or produced results (or that some variables cannot be data because of irreversible algorithms). Feedback from student was generally positive: they pointed out (a) the fact that they can see a path to the problem solution as resulting from the data without carrying out any computation and (b) that the method boosts their confidence that they can do standard-design computations, even if they have no full or previous knowledge of the topic, or if the involved mathematics is not clear to them from the start.

Remarks and Future Research

The author plans to further use the FPD in his sophomore design class. A study is under planning and it will be conducted during the spring semester of the year 2005 on the compared performances of students when solving problems with and without the help of the FPD methodology. Each student will attempt solving, without knowledge of FPD, assigned problems from a set. The same student group will then be introduced the technique and each student will be assigned different problems from the set. For each of the considered problems there will be data about attempting the problem without previous knowledge of FPD (i.e., the "control data") and with the use of FPD (i.e., the "experimental data"). A statistical analysis of such data is under planning; the experimental design and appropriate metrics are under discussion at time of writing this paper. The study will also include a standard questionnaire of student opinions on the usefulness of the method; the questionnaire is included in this paper Appendix.

The author also plans to investigate other uses of the FPD methodology in engineering problem solving: It is interesting to note that the FPD may provide a graphics way of checking data sufficiency and/or data redundancy, but this property has not yet been investigated.

Acknowledgements

The author gratefully acknowledges the introduction to the FPD idea and encouragement to carry out this work from Dr.G.Kardos, Carleton University, Canada, suggestions from Dr.R.Vidal, Technical University of Denmark, and useful criticism from anonymous reviewers.

Bibliographic Information

[1] Raviv, D., Do We Teach Them How to Think?, Proceedings of the 2002 American Society of Engineering Education Annual Conference and Exposition, Session 2630.

[2] Little, P., Cardenas, M., Use of "Studio" Methods in the Introductory Engineering Design Curriculum, ASEE Journal Of Engineering Education, Vol.90, No3, 2001, 309-318.

[3] Ullman, D., The Mechanical Design Process, McGraw-Hill, New York, NY, 2003.

[4] Klebanoff, A.D. and Winkel, B.J., Technology-based problems in calculus from science and engineering, Proceedings of the 1996 American Society of Engineering Education Annual Conference & Exposition, June 1996, Session 1265.

[5] Clark, G., Mitra, A., Molina, G., Effects of Variable Mix on Student Mathematics Performance, Proceedings of the 2004 American Society of Engineering Education Annual Conference & Exposition, June 2004, Session 3565.

[6] Novak, G. S., Diagrams for Solving Physical Problems, in Diagrammatic Reasoning: Cognitive and Computational Perspectives, Janice Glasgow, N. Hari Narayanan, and B. Chandrasekaran editors, AAAI Press / MIT Press, Boston, MA, 1995, 753-774.

[7] Larkin, J., Simon, H. A., Why a Diagram is (Sometimes) Worth 10,000 Words, Cognitive Science, Vol.11, 1987, 65-99.

[8] Fuller, M., Collected Papers of the ERM Division ASEE, 1975, 114-128.

[9] Polya, G., How to Solve It?, Princeton University Press, Princeton, New Jersey, 1945.

[10] Kardos, G., Problem Solving with the Fuller-Polya Diagram, International Journal of Applied Engineering Education, Vol.1, No.2, 1985, 105-111.

[11] Hicks, T.G., Mechanical Engineers Formulas Pocket Guide, McGraw-Hill, New York, NY, 2003.

[12] Wales, C. E., Stager, R. A., Thinking with equations: Problem solving in math and science, Franklin KY: The Writers Block, Center for Guided Design, West Virginia University, Morgantown, 1990.

[13] Vidal, R.V.V., Becker, P.W., On Lagrange Multipliers in Work with Quality and Reliability Assurance, IEEE J. on Selected Areas in Communication, Vol.SAC-4, No.7, 1986, 1143-1148.

[14] Shigley, J.E., Mischke, C.R., Budynas, R.G., Mechanical Engineers Design, McGraw-Hill, New York, NY, 2003.

Biographical Information

DR. GUSTAVO J. MOLINA obtained a Diploma in Mechanical and Electrical Engineering from National University of Cordoba, Argentina, in 1986. Until 1992 he acquired expertise in PVD coatings, vacuum techniques and the design of experimental equipment. In 1994 he received a Master's degree in Mechanical Engineering from the University of Ottawa, Canada, where he conducted research on nondestructive characterization of low-energy impact properties of polymers and composites. He obtained a Doctorate degree in 2000 from Virginia Tech for his work on characterization of electron triboemission from ceramic surfaces. Dr. Molina is currently Assistant Professor of Engineering Studies at The Allen B. Paulson College of Science and Technology, Georgia Southern University, where his current teaching interests include creative design, problem-solving techniques and the development of studio methods in Engineering. He also teaches graphics communication, CAD and computer applications in Engineering.

Appendix

I. Computation presented in Figure 2: Pump shaft design [See reference 11, pages 272-273]

(i) Torsional stress S_S is computed:

For solid shafting: $S_S = (16\ T) / D_0^2$

For tubular shafting: $S_S = (16\ T) / (D_0^2 (1 - (D^4 - D_0^4)))$

Where:

- S_S : torsional stress
- T : transmitted torque
- D : shaft inner diameter
- D_0 : shaft outer diameter

(ii) Critical speed N_{crit} is computed from:

(ii.a) Shaft deflection y under own weight:

$$y = (5\, w\, L^4) / (384\, E\, I)$$

Where:

w : shaft own weight per unit length
L : shaft length between supports
E : Young modulus
I : moment of inertia for solid shafting: $I = 3.1416\, D_0^4 / 64$
and for tubular shafting: $I = 3.1416\, (D_0^4 - D^4) / 64$

(ii.b) Then critical speed N_{crit} in rpm:

$$N_{crit} = 187\, (1 / y)^{1/2};$$ for y in inches.

II. Computation presented in Figure 3: Flat open belt-drive computation [See reference 13, page 871]

(i) Contact angles O_d and O_D (for respectively small and large pulley) are computed:

$$O_d = 3.1416 - 2 \arcsin((D - d) / (2\, C))$$

$$O_D = 3.1416 + 2 \arcsin((D - d) / (2\, C))$$

Where:

D : diameter of large pulley
d : diameter of small pulley
C : center distance

(ii) Then belt length L is computed:

$$L = (4\, C^2 - (D + d)^2)^{1/2} + 0.5\, (D\, O_D + d\, O_d)$$

III. Standard questionnaire for ongoing and future study.

<u>Anonymous Student Evaluation of the Fuller-Polya Diagram (FPD) idea</u>
Do Not write your name on this page.

Indicate your level of agreement with the following items:

	Strongly disagree	Disagree	Unsure	Agree	Strongly agree
1. With**out** the knowledge of FPD, the assigned problem was difficult to solve.	1	2	3	4	5
2. By using the FPD, the assigned problem was easier to solve.	1	2	3	4	5
3. The FPD increased my understanding of the problem.	1	2	3	4	5
4. I would be difficult to solve some problems without the help of the FPD.	1	2	3	4	5
5. At times the FPD idea may be confusing.	1	2	3	4	5
6. I enjoyed using the FPD.	1	2	3	4	5
7. FPD helped me see the whole picture for the problem.	1	2	3	4	5
8. FPD is an appropriate topic for the class.	1	2	3	4	5
9. I look forward using the FPD in the future.	1	2	3	4	5
10.Using the FPD increased my understanding of standard computations in the practice.	1	2	3	4	5

Please list any advantages of learning and/or using the FPD idea.
____
____
____

Please list any **dis**advantages of learning and/or using the FPD idea.
____
____
____

What could be done to make the FPD a better learning experience?
____
____
____