EnviroInfo 2013: Environmental Informatics and Renewable Energies
Copyright 2013 Shaker Verlag, Aachen, ISBN: 978-3-8440-1676-5

Requirements Prioritization Framework for developing Green and Sustainable Software using ANP - based Decision Making

Sedef Akınlı Koçak[1], Giovanna Gonzales Calienes[1], Gülfem Işıklar Alptekin[2], Ayşe Başar Bener[3]

Abstract

Sustainable software production is a relatively new research area within green IT. Software development industry has started getting pressure from regulators and their customers to consider green software development. They often need to make a trade-off among different requirements based on certain criteria and company objectives. This research evaluates environmental sustainability and software quality criteria using a well-known multi criteria decision making approach: Analytical Network Process (ANP). The aim is to prioritize green software criteria in order to use in trade-off models. ANP involves identification of the interrelationship and the intensity of importance and influence between different criteria. The results are presented to be useful in energy saving models of software and/or software development process.

1. Introduction

Software plays an increasing role in our lives and its sustainability and environmental impact becomes more important. Thus green requirements of software need to be considered in addition to traditional quality requirements in software development. In understanding of green and sustainable software Neumann et al. (2011) define green software and sustainable software as *"Green and Sustainable Software is software, whose direct and indirect negative impacts on economy, society, human beings, and environment that result from development, deployment, and usage of the software are minimal and/or which has a positive effect on sustainable development*" (Neumman et al. 2011, pp:296). In this sense green and sustainable software highly depends on both having a green software development process and having a purpose of green software usage.

Software development companies operate under heavy competitive pressure to produce products with more/ better functionality than rivals and with high quality. Sometimes they have to make trade-off decisions in between functional and non-functional requirements (Ruhe and Greer, 2003). Therefore, different stakeholders and different product requirements have to be taken into account. Requirements prioritization and release planning have been made for a successful delivery of the software product. In the literature, there are various approaches and models for requirements prioritization (Karlsson et al. 1998). In these studies the analytic hierarchy process (AHP), ANP/AHP combining method, binary search tree creation, greedy-type algorithms are commonly used methods. A well known approach on software engineering decision support was proposed by Ruhe et.al (2003). They consider requirements prioritization based on the factors such as customer satisfaction, resource allocation and business value/cost using optimization method by using an iterative and genetic algorithm (Ruhe et.al, 2003; Ruhe/Ngo-The 2009).

[1] Environmental Applied Science and Management, Ryerson University, Toronto, Canada, sedef.akinlikocak@ryerson.ca; giovannasalome.gonza@ryerson.ca

[2] Computer Engineering, Galatasaray University, Istanbul, Turkey, gisiklar@gsu.edu.tr

[3] Data Science Laboratory, Mechanical and Industrial Engineering, Ryerson University, Toronto, Canada, ayse.bener@ryerson.ca

Green and sustainable software development has recently become important such that environmental constraints are affecting decision processes. When environmental sustainability is targeted, green software development should trade off between environmental sustainability criteria (e.g., energy consumption) and traditional quality requirements (e.g., usability, performance, security, maintainability). On the other hand, we have not seen any in the literature such that environmental factors are considered with software quality criteria. In this regard, our study is the first attempt to discuss both factors (environment and quality) by using multi-criteria decision-making (MCDM)-based model with ANP. MCDM provides a useful set of tools for understanding trade-offs and gaining insight into alternatives (Bell et al., 2000) in the presence of multiple, usually conflicting decision criteria (Campanella and Ribeiro, 2011). These methods may be used to prioritize the criteria and improve the quality of decision by providing information on trade-offs, increase confidence in decisions and provide insight into the criteria and alternatives (Campanella and Ribeiro, 2011). ANP is one of the most widely used MCDM approaches. It is a generalization of one of the most known MCDM methodologies: Analytic Hierarchy Process (AHP) (Saaty, 2004).

The rest of the paper is organized as follows. Section 2 discusses the concepts of green and sustainable software, environmental criteria with green metrics, software quality models and ANP. In Section 3, introductory information on the ANP methodology is given. The proposed approach of software requirements prioritization and its results are presented in Section 4. Section 5 gives conclusions and discussion about theoretical and practical contributions of this research.

2. Related Literature

2.1 Green and Sustainable Software

Neumann et al., (2011) have stated that green and sustainable software can be achieved as an outcome of a green and sustainable software engineering process. Therefore, the negative and positive impacts of various metrics on sustainable software development need to be studied in order to optimize the software product. One of the pioneering works related to sustainability and software engineering is the reference model "GreenSoft" of Neumann et al. (2011). This model contains a lifecycle of software products, sustainability criteria and metrics, procedures, recommendations and tools that provide suggestions about how to use software in a green and sustainable manner.

In another study, Albertao et al. (2010) relate common software quality attributes (e.g. efficiency, performance, modifiability reusability) to the aspects of sustainability (economic, social and environmental). They also present metrics to quantify each one of the quality aspects. This study did not show how the results should be used to reach the pre-determined effects on sustainability. Recently, two approaches have been introduced related to sustainability and software quality. The first approach considers sustainability as a new factor that affects software quality (Calero/Barteo 2013). A second approach of quality model for green and sustainable software is presented by Kern et al. (2013). Their model helps to decide if software is green and sustainable or not. The model covers all criteria that are directly and indirectly related to software sustainability. The authors have mentioned that the quality aspects should be prioritized for an efficient decision making process. Hence, this has shaped our motivation in this research. We aim at prioritizing both software quality and environmental sustainability criteria, in order to help decision makers.

2.2 Green Metrics

Green metrics are the criteria to quantify the green performance of IT systems. According to Kipp et al. (2011), green metrics are energy consumption-related metrics. The authors have stated that these indicators are the key drivers to identify the "greenness" of an IT application and to indicate the energy consumption, energy efficiency and energy saving possibilities. They have defined four clusters of metrics based on the Green Performance Indicators (GPI):

- **IT resource usage metrics** that are related to energy consumption of IT resources,
- **Lifecycle metrics** that allow setting applications to monitor energy consumption and develop energy aware indicators,
- **Energy impact metrics** that are related to the lifecycle impact on the environment, including the electricity, the power supply, the consumed material, and the CO_2 emissions, and,
- **Organizational metrics** that consider the assessment of additional costs due to energy-related initiatives.

The following table (Table 1) summarizes the main categories of metrics used for each cluster (Kipp et al., 2011; Mahmoud/Ahmad 2012).

Table 1. Green metrics classification

IT resource usage	Lifecycle	Energy Impact	Organizational
- IT equipment utilization / efficiency - IT equipment space / efficiency	- Lifecycle cost - Process engineering - Quality of service	- Power/energy (e.g. green energy usage) - Thermal and humidity - Greenhouse gas (e.g. CO_2 emissions)	- Laws and regulations - Resource efforts - Greenhouse gas credits - Total cost of IT - Green solutions (ex: Return of Green Investment)

2.3 Software Quality Models

Software quality is "*the degree to which software possesses a desired combination of quality attributes.*" (IEEE Std 1061, 1998). In order to assess software quality, this desired combination of attributes needs to be clearly defined. Since software development projects commonly have requirements on developing certain features within a certain time and below a certain cost, managers commonly need to make trade-offs between the attributes. A useful resource related to software quality attributes and trade-offs is the research of Berander et al. (2005). The authors have stated that technical contradictions in quality requirements bring in the trade-offs. They concluded that quality models actually reduce the notion of quality to a few relatively simple and static attributes. Therefore, benefits of quality modelsare that they are simpler to use and they help to identify trade-offs.

Although there are several quality models in the literature, and they consider different quality attributes, the most relevant ones are summarized in Table 2 (Marciniak, 2003; Berander et al., 2005).

Table 2. Quality models in literature

(McCall, 1977)	(Boehm et al., 1978)	(Bowen et al., 1985)	(Murine, 1983)	Others
		Correctness		Correctness
Correctness	Reliability	Reliability	Correctness	Reliability
Reliability	Efficiency	Efficiency	Reliability	Efficiency
Efficiency	Human	Usability	Efficiency	Usability
Usability	Engineering		Usability	
		Integrity		Integrity
Integrity	Understandability	Maintainability	Integrity	Maintainability
Maintainability	Modifiability	Flexibility	Maintainability	Flexibility
Flexibility	Testability	Variability	Flexibility	Testability
Testability	Portability	Portability	Testability	Portability
Portability		Reusability	Portability	Reusability
Reusability		Interoperability	Reusability	Interoperability
Interoperability		Survivability	Interoperability	Survivability
				Safety
		Expandability	Interoperability	Manageability
				Functionality
				Supportability

Based on the models of McCall (1977) and Boehm et al. (1978), ISO has released the standard ISO/IEC 9126 (ISO/IEC 9126-1, 2001). It is an international standard for the evaluation of software. The standard aims to consider well-known human biases that may adversely influence the delivery and perception of a software development project. The standard is divided into four parts in terms of the addressing issues: quality model, external metrics, internal metrics, and quality in use metrics. ISO 9126-1 proposes that the end users experience quality through four factors related to quality-in-use (functionality, reliability, usability, and efficiency), and two factors related to developer-oriented attributes (maintainability and portability) (Fig. 1).

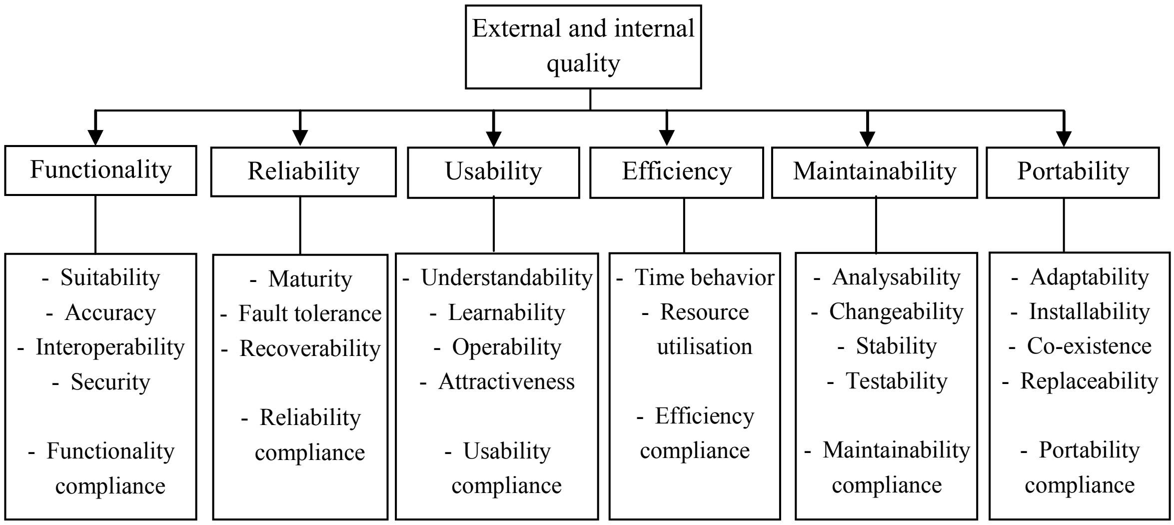


Figure 1.
ISO 9126-1 Quality Factors

3. Analytical Network Process (ANP) Methodology

ANP model that is the generalization of the Analytical Hierarchy Process (AHP) developed by Saaty (1980; 2004) deals with dependency and feedback among clusters. AHP is limited to relatively static and unidirectional interactions with little feedback among decision components and alternatives. ANP provides a general approach to deal with decisions considering the dependence of higher level elements to lower level elements and the dependence of the elements within a level.

ANP framework has three basic features (1) defining the goal, criteria and sub-criteria (2) defining the interdependencies and the network, and (3) building the supermatrix and synthesizing (Gürbüz et al., 2012). The pairwise comparisons between criteria and sub-criteria are performed using Saaty's 1-9 scale (Table 3), which considers the relative importance/ influence of one element over another (Saaty, 2004). The ANP requires that the consistencies of pairwise comparison matrices are checked to prove the consistency of comparisons (Saaty, 1980). Then, the composite weights are extracted considering the interrelationships between criteria and sub-criteria. The supermatrix, which involves these composite weights, is transformed to a weighted supermatrix where each column sums to unity. The last step consists of rating the alternatives in respect to all criteria and sub-criteria, to compute the overall performance score of each alternative. In our research, our alternatives are green and non-green software; hence we do not define them as alternatives. We utilize ANP approach, only to define priority weights of the green metrics.

Table 3. Fundamental scale used when making judgments

Intensity of importance	Definition	Explanation
1	Equal importance	Two activities contribute equally to the objective.
3	Moderate importance	Experience and judgment slightly favor one activity over another.
5	Strong importance	Experience and judgment strongly favor one activity over another.
7	Very strong importance	An activity is favored very strongly over another, its dominance demonstrated in practice
9	Extreme importance	Evidence favoring one activity over another is of the highest possible order of affirmation.

4. Application of the ANP Approach

4.1 Model Determination

The procedure of the prioritization of subjective criteria consists of three steps:

1. Identifying the criteria and sub-criteria.
2. Building of the ANP model.
3. Analyzing the interdependencies between the criteria and sub-criteria of the same cluster.

Each step is presented in more detail in the following sub-sections.

4.1.1 Determination of criteria and sub-criteria

Major criteria related to sustainability of a software product are determined in respect to the related literature (see sub-section 2.3 and 2.4). Then, several criteria that have direct relationships with major criteria are selected as sub-criteria. Table 4 summarizes all the criteria and sub-criteria that have been used to construct ANP model.

Table 4. Software quality and environmental sustainability criteria

Software Quality Criteria (QC) (ISO / 9126-1, 2001)

QC_1	Functionality	System resources used to achieve required software functionality.
QC_2	Reliability	Totality of essential functions that the software product provides.
QC_3	Usability	Capability of the system to maintain its service provision under defined conditions for defined periods of time.
QC_4	Efficiency	Ease of use of a given function.

Environmental Sustainability Criteria (EC) (Mahmoud/Ahmad 2012)

EC_1	Energy Consumption	Total electricity consumption during operation.
EC_2	CO_2 Emission	Amount of average carbon dioxide emissions.
EC_3	Green Energy Usage	Usage of renewable energy.
EC_4	Return of Green Investment	Time it takes for green solutions to pay off or recuperate.

4.1.2 ANP Framework

The framework of our ANP model illustrates the interactions among goal and criteria and sub-criteria (Fig. 2). The proposed decision model consists of two levels. The objective (developing green and sustainable software) is situated at the first level. In the second level, the criteria are listed. In the proposed hierarchy, outer dependencies and inner dependencies among sub-criteria as well as interaction between criteria are assumed to be present. The Interdependencies and outer dependencies are shown in arrows.

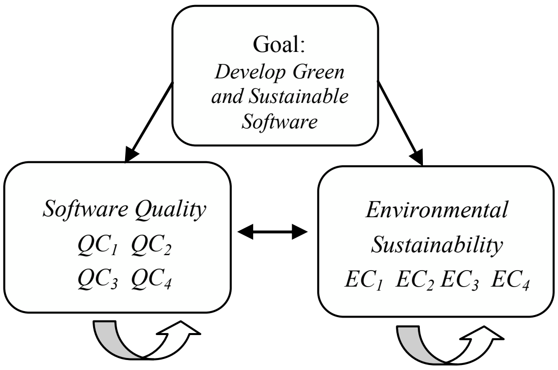


Figure 2.
Framework and Interactions among Criteria

4.1.3 Analysis of the interdependencies between the criteria and the sub-criteria

As the last step of the ANP methodology, pairwise comparison matrices for all the sub-criteria and criteria have been prepared as a questionnaire. The questionnaire was filled by ten decision makers (DM) from four different software development companies. Each DM is an expert in the software engineering field. The geometric mean of 10 questionnaires is used to form the output of the ANP approach (Saaty/Vargas 2006). The consistency indexes of all these matrices are found to be smaller than 0.10. The pairwise comparisons enable us to retrieve relative weights for the sub-criteria.

Table 5. Pairwise comparison matrix of two major criteria

	EC	QC
EC	1	6
QC	1/6	1

Table 6 and Table 7 represent the geometric means of the pairwise comparisons between software quality related sub-criteria and environmental sustainability related sub-criteria, respectively.

Table 6. Pairwise comparison matrix of software quality related sub-criteria

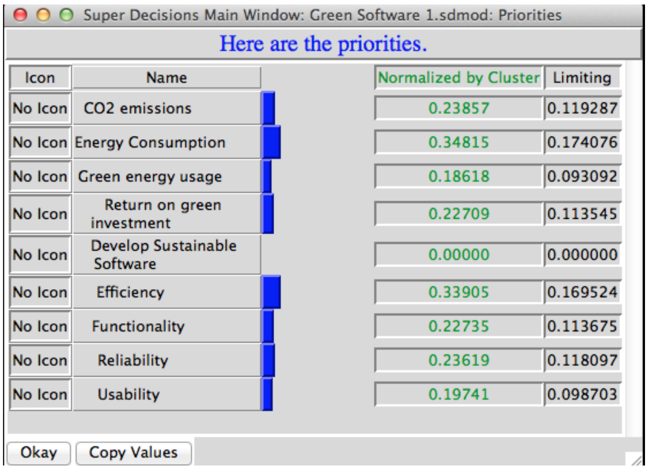
	QC_1	QC_2	QC_3	QC_4
QC_1	1	1/3	1/2	1/2
QC_2	3	1	4	3
QC_3	2	1/4	1	1/3
QC_4	2	1/3	3	1

Table 7. Pairwise comparison matrix of environmental sustainability related sub-criteria

	EC_1	EC_2	EC_3	EC_4
EC_1	1	2	2	1
EC_2	1/2	1	2	1/2
EC_3	1/2	1/2	1	1
EC_4	1	2	1	1

Let us look at the pairwise comparison matrix of major two criteria (Table 5). Hence, the DM has to decide which of the two criteria sets, software quality and environmental sustainability, is more important with respect to develop a green and sustainable software product. The geometric means of the responses are taken in terms of Saaty's scale of measurement (Table 3), in order to determine aggregate individual judgments.

We have used the Super Decisions software product (Creative Decisions Foundation, 2013), in order to calculate the results of final priorities of the eight sub-criteria (Fig. 3). Therefore, the environmental sustainability criterion is found as the most preferred one to achieve to develop green and sustainable software. The results indicate that the energy consumption criterion has the highest priority (35% among the EC criteria set). It is followed by software efficiency (34% among SQ criteria set). Software efficiency considers how much information technology resources are used efficiently in terms of energy. The third most important criterion is found as the CO_2 emission with a weight of 24% among EC criteria set. This is closely related to energy consumption.



Super Decisions Main Window: Green Software 1.sdmod: Priorities

Here are the priorities.

Icon	Name	Normalized by Cluster	Limiting
No Icon	CO2 emissions	0.23857	0.119287
No Icon	Energy Consumption	0.34815	0.174076
No Icon	Green energy usage	0.18618	0.093092
No Icon	Return on green investment	0.22709	0.113545
No Icon	Develop Sustainable Software	0.00000	0.000000
No Icon	Efficiency	0.33905	0.169524
No Icon	Functionality	0.22735	0.113675
No Icon	Reliability	0.23619	0.118097
No Icon	Usability	0.19741	0.098703

Okay Copy Values

Figure 3.
Final Priorities of Cluster Elements

5. Conclusion and Future Work

Most software development companies need to adapt their development processes to meet new software business standards and demands. One of the fundamental demands is to develop environmentally friendly products. In many occasions, these objectives are achieved at a cost of reducing or increasing the consumption of different resources within the organization. As a result, the software development industry needs making certain trade-offs between the demands of end-users and the requirements for sustainability. This research proposes an approach for software requirements prioritization based on a multi-criteria decision-making model. A well-known MCDM technique, the ANP, is used as the framework. The framework has defined the weights of the sub-criteria. These weights will enable DMs to make more accurate decisions when dealing with trade-off models. This framework could be applied to any domain of activity to prioritize environmental concerns in any type of decision making process. Further work will be conducting an empirical work in a real life software project. Doing so, we will create our goal model and use obtained weights from the requirement prioritization framework as weight coefficients.

Bibliography

Albertao, F., Xiao, J., Tian, C., Lu, Y., Zhang, K. Q., & Liu, C. (2010): Measuring the sustainability performance of software projects. In e-Business Engineering (ICEBE), 2010 IEEE 7th International Conference on (pp. 369-373). IEEE, pp. 369–373.

Bell, M.L., Hobbs, B.F., Elliott, E.M., Ellis, H., Robinson, Z. (2000): An Evaluation of multicriteria decision-making methods in integrated assessment of climate policy, in: Research and Practice in Multiple Criteria Decision Making, Lecture Notes in Economics and Mathematical Systems, 487, pp. 228-237.

Berander, P., Damm, L., Eriksson, J., Gorschek, T., Henningson, K., Jonsson, P., Kagstrom, S., Milicic, D., Martensson, F., Renkko, K., and Tomaszewski, P. (2005): Software quality attributes and trade-offs. Blekinge Institute of Technology.

Boehm, B.W, Brown, J.R., Kaspar, J.R., et al., (1978): Characteristics of Software Quality, TRW Series of Software Technology, Amsterdam, North Holland.

Bowen, T.P., Wigle, G.B., Tsai, and J.T. (1985): Specification of software quality attributes, Techical Report RADC-TR-85-37, Rome Air Development Center.

Calero C. and Bertoa, M.F. (2013): 25010+S: A software quality model with sustainable characteristics. Sustainability as an element of software quality. Green in Software Engineering Green by Software Engineering (GIBSE 2013) co-located with AOSD 2013, March 2013. Fukuoka, Japan.

Campanella, G., Ribeiro, R.A. (2011): A framework for dynamic multiple-criteria decision making, in: Decision Support Systems, 52, pp. 52–60.

Creative Decisions Foundation. (2013): Super Decisions (ver. 2.2.6) [Open source Software]. Retrieve February 2, 2013. Available from: http://www.superdecisions.com.

Gürbüz, T, Alptekin, S.E. and Işıklar Alptekin, G. (2012): A hybrid MCDM methodology for ERP selection problem with interacting criteria, Decision Support Systems, 54, pp. 206–214.

IEEE Std 1061-1998 (1998), IEEE Standard for a Software Quality Metrics Methodology, IEEE Computer Society.

ISO/IEC 9126-1 (2001): Information technology - Software quality characteristics and metrics - Part 1: Quality characteristics and subcharacteristics, International Organization for Standardization, International Electrotechnical Commission, Geneva.

Karlsson, J., Wohlin, C., and Regnell, B. (1998): An evaluation of methods for prioritizing software requirements, Information and Software Technology, 39, pp. 939–947.

Kern, E. Dick, M., Naumann, S, Guldner, A, and Johann, T. (2013): Green Software and Green Software Engineering – Definitions, Measurements, and Quality Aspects, In Proceeding of the First International Conference on ICR for Sustainability (Zurich), pp. 87-94.

Kipp, A., Jiang, T., Fugini, M. (2011): Green metrics for energy-aware IT systems, IEEE Computer Society, 42, pp. 241-248.

Mahmoud, S., Ahmad, I. (2012): Green performance indicators for energy aware IT systems: Survey and assessment, in: Journal of Green Engineering, 3, pp. 33–69.

Marciniak, J.J. (2003): Encyclopedia of Software Engineering, 2003, Wiley.

McCall, J.A. (1994): Quality Factors, Encyclopedia of Software Engineering, John Wiley & Sons.

Murine, G.E. (1983): Improving management visibility through the use of software quality metrics, IEEE Computer Society's 7th International Computer Software & Applications Conference, pp. 638-639.

Naumann, S., Dick, M., Kern, E., Johann, T. (2011): The GREENSOFT Model: A reference model for green and sustainable software and its engineering, in: Sustainable Computing: Informatics and Systems, 1, pp. 294-304.

Ruhe G, Greer, D. (2003): Quantitative studies in software release planning under risk and resource constraints. In Proceedings of International Symposium on Empirical Software Engineering (ISESE), IEEE, Los Alamitos CA, pp. 262-271.

Ruhe, G. and Ngo-The, A. (2009): Optimized resource allocation for ssoftware release planning. Trade-off analysis for requirements selection. IEEE Transactions on Software Engineering, 35, pp.109–123.

Ruhe, G., Eberlein, A., and Pfahl D. (2003): Trade-off analysis for requirements selection. International Journal of Software Engineering and Knowledge Engineering, 13, pp. 345–366.

Saaty, T.L. (1980): The Analytical Hierarchy Process. New York: McGraw Hill. Pittsburgh: RWS Publications, 4922 Ellsworth Avenue, Pittsburgh, PA 15213.

Saaty, T.L. (2004): Fundamental of the analytic network process-Dependence and feedback in decision-making with a single network. Journal of Systems Science and Systems Engineering, 13 (2), pp. 129-157.

Saaty, T.L., and Vargas, L. (2006): Decision making with the Analytic Network Process: Economic, political, social and technological applications with benefits, opportunities, costs, risks. Springer, NY.

Silveira, G.D., and Slack, N. (2001): Exploring the trade-off concept, International Journal of Operations & Production Management, 21(7), pp. 949-964.