Integrating Problem Solving and Research Methods Teaching for Systems Practice in Engineering

Mike Yearworth\textsuperscript{a,*}, Gordon Edwards\textsuperscript{a}, John Davis\textsuperscript{a}, Katharina Burger\textsuperscript{a}, Adrian Terry\textsuperscript{b}

\textsuperscript{a}Systems Centre, Faculty of Engineering, University of Bristol, BS8 1TR, UK
\textsuperscript{b}Thales Training and Consultancy, Crawley, West Sussex, RH10 9XN, UK

Abstract

Problem solving and research methods apparently sit within different traditions of development evidenced by disparate sources of literature. However, in the graduate education of engineers taking an Engineering Doctorate (EngD) Program in Systems there is a need for their integration in such a way as to make their relationship clear. We argue from experience of course delivery and project supervision that research methods from business and management need to support a generic problem solving approach – informed from the Problem Structuring Methods (PSM) literature, and specifically Soft Systems Methodology (SSM) – such that they provide the rigorous evidence needed at any stage of a problem solving cycle. There is a clear hierarchy with a problem solving approach providing the guiding methodology for systems practice in engineering, and research methods supplying the means to generate answers to specific questions as they arise. We specifically discuss the special role of action research as both a problem solving and a research strategy and its relevance to engineering education, and suggest a philosophical underpinning for the approach.

1. Introduction

The Engineering Doctorate in Systems program has been offered since 2006 by the Industrial Doctorate Centre (IDC) in Systems, a joint activity between the Engineering Faculty at the University of Bristol and the Management School at the University of Bath. The program is aimed at high-caliber engineers, from recent graduates through to early/mid-stage career level. The origins of the Engineering Doctorate (EngD) in the UK as an alternative to a PhD for graduate-level engineering education are described in the Parnaby Report commissioned by the Science and Engineering Research Council (SERC) in 1990 [1].
As of October 2012 there have been 96 Research Engineers recruited to this 4-year program with sponsoring organizations ranging from small to medium-sized enterprise (SMEs) through to multinationals. Projects span a number of industry sectors ranging from construction, water industry, energy, rail, aerospace, and defense. Projects are initiated and sponsored by industry and Research Engineers are recruited to the program in order to undertake a program of research that will fulfill the needs expressed in the initial problem statement presented by the sponsoring company. This presents the formative challenge of simultaneously satisfying industrial needs and delivering a doctoral level contribution to knowledge [1].

Teaching is based on methods drawn from systems engineering, business and management, and systems thinking and aims to integrate them into a coherent program. Pedagogical development emerges from the process of learning together on the program; it is driven by the systemic nature of the enquiry into real-world complex problem situations. This paper is focused on the research methods and problem solving module and discusses how we have arrived at a means of integrating them into a coherent structure to support the objectives of the overall program.

2. The Existing Problem Solving Approach

Current teaching on the program introduces problem solving in a generic way by using the idea of a basic research process tailored for an EngD in Systems and consisting of four processes

• Problem investigation and exploration
• Methods and methodology
• Findings
• Action and intervention

Since research in its most general form is a process of enquiry there is a multiplicity of potential purposes of research in relation to a problem situation described as a system and ranges on a spectrum from curiosity driven to applied, or needs-based. The EngD in Systems program is intentionally focused on industry needs or problem solving and therefore includes the research design necessary for designing, implementing and trialing interventions and solutions.

We use the roller coaster ride as a metaphor to emphasize how the problem solving processes in complex situations is different, suggesting constant recursive engagements with the problem, trial solutions and maybe only moving slowly to a solution with the possibility of set-backs on the way. The roller coaster nature of a problem solving journey is an interpretation of [2] and is in fact the front elevation or projection in two dimensions of one or more action learning cycles. In using this metaphor we are trying to help the Research Engineers link their existing problem solving experience, gained from industry or undergraduate programs, with the new ideas presented on the program. In this context we also discuss in some detail of what an end-point to a project might be, and of the pragmatic stopping rules which can be used in such an action learning cycle due to the fact that complex wicked problems are usually never solved. However, the actual detailed definition of a complex wicked problem is problematic as a definition would assign to them ontological status and is contrary to the mindset that we are trying to develop on the program. What this means in practice is that the creative tension which may exist due to different problem frames can inspire a process of synthesis whereby the knowledge claims [3] underpinning each problem frame are challenged by the Research Engineers, industry stakeholders and academic supervisors such that an awareness of their respective limitations is gained. Through this individual and collective learning process, “action to improve” [4] is enabled and a process of reliable knowledge generation facilitated [5, 6].

Also introduced in the module was a major point of discussion on how to describe and justify this cyclic never-ending journey to sponsors and others who are more used to a waterfall (requirements driven) engineering or a one-shot approach in the knowable and known regions of Kurtz and Snowden’s Cynefin framework [7]. The Research Engineers are left at the end of the module with the learning outcome that research method rigor is crucial to help underpin or evidence a problem solving approach in order to maximize impact and usefulness of the research project to its industrial sponsors. A more detailed explanation for the current structuring of research methods teaching and its relation to systems thinking is presented in an earlier paper [8].

The actual strategies of research methods are well understood. A simple framing device to explain terminology used is based on Figure 4.1 of [9] and is presented in Table 1.
Table 1. An unpeeling into layers of the research onion of [9] to present the terminology used in this paper.

<table>
<thead>
<tr>
<th>Layer</th>
<th>Example Choices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Philosophies</td>
<td>Positivism, Realism, Interpretivism, Pragmatism</td>
</tr>
<tr>
<td>Approaches</td>
<td>Deductive, Inductive</td>
</tr>
<tr>
<td>Strategies</td>
<td>Experiment, Survey, Case Study, Action Research, Grounded Theory, Ethnography,</td>
</tr>
<tr>
<td></td>
<td>Archival Research</td>
</tr>
<tr>
<td>Choices</td>
<td>Mono Method, Mixed Methods, Multi-Method</td>
</tr>
<tr>
<td>Time Horizons</td>
<td>Cross-Sectional, Longitudinal</td>
</tr>
<tr>
<td>Techniques and Procedures</td>
<td>Data collection and data analysis</td>
</tr>
</tbody>
</table>

A number of problem solving approaches have been introduced on the program, either explicitly in research methods teaching, or used behind the scenes to structure the program itself. For example, the creative problem solving approach of [10] has been used as the method for structuring an EngD in a Day workshop and is shown in Figure 1(a). This workshop is designed as an exercise to consolidate research methods learning where teams of Research Engineers are given one of a number of needs-driven problem statements from our industrial partners and are facilitated through this problem solving process to develop a fully-fledged research plan that can be presented back to the industry sponsor at the end of the day.

![Image](image-url)

Fig 1. (a) Problem solving approach, based on CPSB Inc. Creative Problem-Solving Framework 6.1TM, developed by Adrian Terry and derived from [10] used for structuring the EngD in a Day workshop in the module; (b) A system value cycle view expressed as a problem suppression system (PSS) by Jack Ring [11].

We have also recently incorporated ideas from Ring [11], which he has expressed as a value seeking approach to the engineering of systems and is shown in Figure 1(b) and as taken up recently by the joint INCOSE/ISSS working group. We see this as an example of an approach to problem solving coming from core Systems Engineering ideas with resonances to Soft Systems Methodology. We also take into account the strong relationship between problem solving and action research and use various ways of visualizing an action research spiral, such as that shown in Fig 2 (a) derived from [12].

Having i) reviewed the use of [10, 11] as appropriate problem solving approaches, ii) proposed in [13] a new approach based on the experience (gained since [8]) of trying to bridge between research methods teaching and Soft Systems Methodology (SSM) [14-17], and iii) attempted to synthesize a generic problem solving approach that could be used by the Research Engineers on the program, which is discussed in the next section, we are continuing our investigation into the role that a problem Structuring Method (PSM) like SSM could take in meeting our needs; it is well-theorized and has been used extensively outside engineering. Our interpretation of SSM is shown in 2 (b). We discuss what it might take to move towards adopting an approach like SSM in [13]. We have yet to investigate whether other PSMs originating from the Soft OR community [18] might be useful. A recent case study in [19] reveals that there may be widespread, informal, and un-codified use of PSMs like SSM in engineering organizations. This suggests that we should go and look for evidence of their use and attempt to build a relationship...
with the management academe, where the use of these methods is reasonably widespread, in order to strengthen the link back to engineering – SSM after all originated from Checkland’s, and others, attempts to apply Systems Engineering to organizational problems in industry [20]. Taking this route would explicitly acknowledge that all the projects on the program will have action research as their overarching research strategy. Checkland has also claimed that there is no reason why SSM should not be used on hard systems engineering problems [21].

The experience and learning from teaching the existing problem solving approach have been described in [8]. In this paper data collected from the Research Engineers on the program was presented and four broad areas of concern were raised about dealing with the “countercultural and counterintuitive ideas from phenomenological and mixed research paradigms” as follows:

1. Questioning the validity of the results from research methods that are broadly phenomenological in approach,
2. Conducting action research,
3. Discomfort of having to justify qualitative research methods in an engineering company, and
4. Developing the social skills required in order to carry out effective qualitative data collection (e.g. conducting semi-structured interviews), and the research skills required to analyze the data thus collected.

It has been the ongoing reflection on these results and further data collection that has led to the need to develop the program and introduce a more generic approach to problem solving to achieve the desired integration of problem solving and research methods teaching. Data from implementing this new development have yet to be collected and analyzed.

3. Developing a Generic Approach to Problem Solving

From these different approaches we are now using a simple schema for problem solving visualized in 3 (a) and has 4 key stages:

1. Exploring
2. Designing (Planning for Action)
3. Implementing (Taking Action)
4. Monitoring/Learning
Fig 3. (a) A 4-stage problem solving approach as a development of current use in research methods teaching; (b) the project may well loop around the phases either deliberately as in an action research program, or spontaneously due to the nature of the research findings.

The approach clearly has resonances with the Plan-Do-Check-Adjust stages of the Deming cycle. We assume that all projects start from a problem, or equally an opportunity statement, which in effect is Stage 0, and is the initiating event for the instance of the approach and is the problem situation unresolved, or unstructured in Checkland’s language. Each step is executed in turn but there is a degree of fuzzy overlap and/or backtracking between neighboring stages. For example, back-tracking from the end of the Implementing/Taking Action stage would be recognized as the process of Change Management. The problem statement at stage 0 does not remain static, the world moves on, so a developed visualization is as the spiral shown in Figure 3 (b). We assert that research methods can be used at any point in any/all of the stages to answer research questions that arise in this process of problem solving. Each stage has an associated set of processes that can be used to progress the stage, by adopting gerund (verbal nouns indicating action) labels we suggest a process model view of the problem solving approach that is designed to emphasize the fact that the Research Engineer is engaged in a process of enquiry. The approach is owned by the Research Engineer; i.e. our perspective here is that the Research Engineer needs a problem solving approach aligned to a systems perspective and focused on delivering a body of research over the 4 year duration of their project.

3.1. Processes of Problem Solving

The processes likely to be associated with each of the stages are listed in Table 2.

Table 2. Stages of the problem solving approach and associated processes

<table>
<thead>
<tr>
<th>Stage</th>
<th>Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exploring</td>
<td>identifying stakeholders, dealing with worldviews, getting industry supervisor to open doors, suggesting different interpretations of the problem/opportunity, challenging the system boundary, exploring the system, helicoptering, removing barriers, recruiting stakeholders, getting buy-in for the project, visualizing the system</td>
</tr>
<tr>
<td>Designing (Planning for Action)</td>
<td>developing options (optioneering), deciding purpose, modelling, testing, researching, agreeing purpose, studying feasibility, getting agreement, examining cultural acceptability, ensuring an ethical approach, costing, value engineering</td>
</tr>
<tr>
<td>Implementing (Taking Action)</td>
<td>making, building, scaling-up, scaling-out, prototyping, leading and managing change, managing projects and/or programs, managing/monitoring costs, bringing people on board, convincing, communicating, monitoring progress, selling the project, training, running workshops, testing, stage-gating, fire-fighting, creating redundancy, realizing early gains/value, picking low-hanging fruit, managing risk/uncertainty, putting measurement in place, going backwards (re-doing, re-working)</td>
</tr>
<tr>
<td>Learning/ Monitoring</td>
<td>monitoring, evaluating, collecting data, discovering POSIWID*, comparing to designed purpose, answering fit-for-purpose questions, measuring value, answering “have we delivered value?” assessing uptake, assessing buy-in, answering “what have we learned?” making longitudinal studies, communicating, identifying new problems/opportunities, redefining problems/opportunities, deciding on whether to go around again, rationalizing stopping</td>
</tr>
</tbody>
</table>

* POSIWID – “Purpose of System is What it Does” from [6].
This is not an exhaustive list but represents a starting point to suggest the things Research Engineers need to be thinking about at each stage. In the language of the problem solving approach from [10] we see exploring in Stage 1 as aligned with a period of *divergence* prior to *convergence* in Stage 2. In fact this process of divergent thinking followed by convergent thinking style is required at all stages of the approach as the Research Engineer explores issues that will be developed or resolved at the next. The expectation at Stage 2 is that the Research Engineer is required to come up with several intervention/change options to address the problem – but only take one of them through to the next two stages. Therefore, some rationale for the selection and choice between options should be apparent though – even though this may not represent the rigor and detail and the full research approach of the optioneering process. Some selection criteria to move to stage 3 will need to be developed. These could be the high potential to achieve business benefit, the ethical acceptability of the option, its cultural feasibility, or even the one that provides maximum scope for learning. We see this stage as a convergence towards a key decision point on a design, or an action plan, or both, and consistent with a view that options have been rejected to arrive at a view of what needs to be undertaken in the implementation phase. It would be too tidy to assume that any particular project on the EngD in Systems Program will execute all 4 stages in sequence and to completion, and in fact this is likely to be an exception. The other possibilities are clearly executing only part of the cycle, or possibly completing many cycles, or spawning one or more cycles to create a program of research.

### 3.2. Roles of Research Methods and Finding and Developing Useful Research Questions

The core texts for Research Methods knowledge that we use on the module are written for the business and management student typical of an MBA program in the UK [9, 23-25]. We feel they offer a good source of information about research methods relevant to the needs of the Research Engineers on the program and represent a compromise between specialist texts for specific industries in Engineering, e.g. [26], or the general social science *tomes* such as [27]. The first test to decide whether research questions are likely to be useful for a Research Engineer to investigate is to use the criteria in the first column of Table 3. The criteria can be understood as a way of expressing the different motivations/values/problem frames that inform the boundary spanning process, i.e. the EngD in Systems specific multi-view problem framing process. The negotiated assessment of utility against these criteria is in effect the program’s (soft) problem structuring method and relates back to the formative challenge of the EngD in Systems stated in the introduction. This process of negotiation needs to take into account the likely differing worldviews of the stakeholders.

We see research questions originating in different stages of the approach and in different ways and we need the Research Engineers to be aware that this is the case, but we can characterize three cases as follows:

1. The research question is given or implicit in the problem/opportunity statement and could represent a clear *grand-tour* question as well as the purpose of the research,
2. The research questions are a result of the Exploring phase of the approach. The problem/opportunity statement merely points the research engineer in a particular direction,
3. The research questions emerge as necessary or opportunistic at any stage and in any process. These research questions could be small (enabling) or big, in which case could potentially lead to a new instance of the problem solving approach loop or a new research project. Each case is likely to lead to research questions of different utility with respect to the criteria listed in column 1 of Table 3.

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Research questions implicit/given</th>
<th>Research questions result from Exploring stage</th>
<th>Research questions emerge or opportunistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contributing to new knowledge</td>
<td>To be investigated</td>
<td>Becomes a test for good research questions</td>
<td>Becomes a test for good research questions</td>
</tr>
<tr>
<td>Meeting the needs of the industrial sponsor</td>
<td>Almost certainly</td>
<td>Becomes a test for good research questions</td>
<td>Almost certainly, else ignore the research question</td>
</tr>
<tr>
<td>Fulfilling the expectations of an Engineering Doctorate in <em>Systems</em></td>
<td>Unknown at start</td>
<td>A result of the process</td>
<td>Depends</td>
</tr>
</tbody>
</table>

Table 3. Evaluating utility of research questions by source and success criteria.
We want the Research Engineers to be aware that research questions are essentially dynamic in nature. They can suggest others, be dead-ends, split into sub-research questions, be combined, become obsolete or otherwise irrelevant, or remain unanswered. All outcomes are potentially useful knowledge and therefore must be documented as good research practice and considered for writing-up in the dissertation.

We see contribution to new knowledge possible at any/all of the 4 stages of the approach. The really unique nature of the program is that a contribution to knowledge could arise from the process of boundary critique itself and we hold this in high-esteem and essential for the further development of systems practice in engineering [28]. Fulfilling the expectations of an engineering doctorate in systems requires evidence of i) an explicit approach to system boundary critique ii) use of theoretical/methodological pluralism to handle complexity/systemicity, and iii) action for improvement, [29] i.e. what makes an EngD in Systems as opposed to an EngD in anything else.

A number of conflicting needs must be met for a successful outcome and the statement of problem/opportunity at Stage 0 that forms the basis for starting the project will possibly be the source. Learning about systems practice in engineering and disseminating that learning is an essential outcome too. We believe that by meeting this additional need will lead to a diffusion of knowledge into the wider systems community.

3.3. Plurality/Triangulation

We encourage the Research Engineers to see that there is a many-to-one relationship between research paradigm/approach/strategy and research questions thus allowing for and encouraging triangulation. The expectation from the systems criteria of Table 3 is that research questions are multi-disciplined in nature; however we do not expect sub-research questions to be so i.e. we expect a 1:1 relationship between sub-research questions and the paradigm/approach/strategy breakdown listed in Table 1. The specific technique of multimethodology from the PSM literature [30, 31] is covered elsewhere on the program.

3.4. Research Method

Derived from each research question is a research strategy designed to find answers. The unpeeled onion of Table 1 is re-interpreted as a process in Figure 4.

```
<table>
<thead>
<tr>
<th>Question(s)</th>
<th>Method</th>
<th>Findings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paradigm Approach Strategy</td>
<td>Experiment</td>
<td></td>
</tr>
<tr>
<td>Data Collection (time) Analysis Interpretation</td>
<td>Modelling Survey Case Study Grounded Theory Ethnography</td>
<td></td>
</tr>
<tr>
<td>Selection/Choosing</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Fig 4. Research Method as a Process.

Modelling and action research are called-out explicitly in Figure 4 for the following reasons:

1. Action Research: we make the point below that given certain conditions the problem solving approach we describe is action research, not just a strategy for answering a research question. This has implications for the way in which the research is conducted, especially what constitutes data, and how the research is written up.

2. Modelling: data may be generated from the results of simulation rather than collected. That this is a valid method is argued for strongly by [32, 33]. The philosophy of this is covered in a module of the program that addresses modelling and simulation so that it is established as a highly valid research strategy early in the program.
3.5. Action Research and Action Learning

Whilst we have shown action research as a research strategy in Table 1, the relationship between action research as a problem solving approach in its own right and action research merely seen as a research strategy is an important one to explore. In our formulation, if a research question is such that it requires one or more iterations of the cycle to satisfy the needs of the sponsoring organization then to all intents and purposes the Research Engineer has been engaged in action research; the four phases of the approach shown in Figure 3 (a) and the associated processes listed in Table 2 present a schema for action research interpreted as a learning system. This would accord with Checkland’s view that SSM is action research although we are aware that there are conflicting views on this e.g. [34]. Action research is still likely to generate research questions and will require the process outlined in §3.4 to be followed. We are aware that action research is little known in systems engineering although it may be possible to learn across from its use in information systems [35-37], however it is clear that more work is necessary to better understand the role that action research plays.

3.6. Philosophy and Ethics

The philosophical underpinnings of the module are based on Dewey’s paradigm of rationality as inquiry with a moral reasoning underpinning [38, 39]. Practical rationality can be described as (a) the process of enquiry (b) taking place in a problematic situation, (c) issuing in a judgment of practice, (d) operating in an articulative way, and (e) aiming at a transformative result [40]. In other words, practical rationality may be created in a deliberative, participatory process whereby the problem solver (the agent) articulates his view and considers views of other participants in the situation; he may transform his own view and possibly the views of others and arrive at shared moral judgment [38]. These forms of reasoning are mirrored in soft systems methodologies [41] which use modelling to support the processes of articulation and transformation. The use of such methodologies in their role as double-loop learning systems [42] can be understood as a way of arriving at ethical judgment or practical rationality. However, in the context of the EngD in Systems program, the process of systemic problem structuring and the creation of negotiated knowledge [43] take place within small learning sets [44]. A localized and contextualized process [29, 45, 46] of problem framing may occur and include normative boundary judgments [29]. Yet, the level of criticality and the transformative potential of the process of boundary critique depend upon the extent of divergence of values and beliefs between the members of the learning sets. In other words, there is a risk that “explicit and implicit standards, conventions, rules and discourse-practices” [47] of the organizational culture that participants and the problem solver are embedded in, are accepted without reflection. Such practice may result in “instrumentally and functionally reproducing accepted meanings and conventional organizations, institutions, and ways of doing things” [47]. In order to facilitate the transformative process of challenging the status quo [47] and enabling the development of practical rationality in the context of systems practice in engineering, an underpinning of the discursive process by Critical Systems Heuristics (CSH) [48, 49] may be suggested. Thus, it is in providing structure to the process of developing ethical judgment through an articulated and transformative process of learning together, that systems methodologies are particularly valuable for pragmatist systems practice in engineering.

4. Conclusions

We have discussed the integration of research methods and problem solving for the development of a module in research methods teaching on the EngD in Systems program offered jointly by the Universities of Bristol and Bath. This has been based on the experience of the authors in delivering the module and in supervision of Research Engineers on the program. It is thus reflective in its form and part of the ongoing learning process as we develop our own distinctive (soft) problem structuring method for systems practice in engineering and is driven by the nature of the formative challenge that the EngD in Systems program creates. Breaking delivery of the module into two parts separated by approximately 6 months allows time for Research Engineers to reflect on problem solving and researching systems in the context of their own project and organization. Focusing the first part of the course on introducing a problem solving approach and researching systems establishes methodology, and the second on reviewing initial exploration of the system and research design provides a check on approach and further
opportunity for group learning and experience in challenging the choice of boundary

Evidence from consultative meetings, conversations, annual program reviews and our own deliberations has pointed us towards more emphasis on clarity on defining a problem solving approach for Research Engineers and then on how research methods fit into problem solving acknowledging a systems context.

Throughout this paper and in the title we have used the term problem solving whereas we would have preferred to have used problem structuring for the reason that when dealing with wicked problems they are never solved as such. Solving tends to suggest action leading to a solved problem whereas structuring suggests a way of engaging with problems, a process of enquiry, and indeed this is precisely what is intended by the label Problem Structuring Methods (PSMs). However, this label is in a sense owned by the Soft OR community, which exists mainly in Europe, and refers to a specific set of PSMs such as Soft Systems Methodology (SSM), Strategic Options Development and Analysis (SODA), Strategic Choice and Viable Systems Model (VSM) amongst others – see [18] for a recent review and critique. We wanted to avoid this confusion in our paper and have stuck doggedly to solving. However, there is an important relationship between the approach we suggest and PSMs such as SSM. SSM is a valid action research approach that could be used by any project on the program [21]. However we have stepped back from advocating its use because we are concerned about its cultural fit in more mainstream engineering practice but this is not a final position. Work by two of the authors (Yearworth and Edwards) focused on what it would take to move to the adoption of a PSM, such as SSM, as the approach for systems practice in engineering [13]. Recent work in systems architecting for systems engineering also points in this direction [50].

Acknowledgements

The authors would like to thank colleagues and Research Engineers on the EngD in Systems program for the myriad experiences, reflections, and stories that have contributed to the richness of this paper. We would like to thank Jack Ring and Janet Singer for providing Figure 1 (b) and useful discussions at the joint INCOSE/ISSS workshop held in San Jose on the 15th July 2012. We would also like to thank Dr. Oksana Kasyutich, Industrial Doctorate Centre (IDC) in Systems Manager, and Research Engineers Charlotte Dunford, Per Saunders and Joe Clarke for their specific feedback about developing research methods teaching. The IDC in Systems is supported by EPSRC grant (EP/G037353/1). We would also like to thank the two anonymous referees for providing suggestions for improving the paper.

References


