Chapter 15
Mechatronics Education: Meeting Future Need

David Russell

15.1 Introduction

Of all the topics feverishly discussed in staff meetings, conferences and blogs, future educational effectiveness and relevance always seems to crop up. Professors, remembering their own education and struggles to attain their current status in academe bemoan the current lack of mathematics, the lack of student talent, and the drift away from hard design to a plug-and-play mentality. There are evident disjoints between Software-as-a-Sservice (SaaS), cloud computing, and Platform-as-a-Service (PaaS) and mechatronic systems.

While abstraction from the internals of the computational units that constitute the Internet of Things (IoT) might speed up product launches, the mechatronic engineer who is operating in the actual application domain is left pondering the integrity of the software and its source, its ruggedness over time in the real world setting, how to manage component upgrades, and the recovery of the system after a failure. This chapter includes true vignettes, disasters, challenges and discussion topics from the author's experience, selected to highlight what a mechatronics engineer must know and to illustrate the necessity for innovation and technical dexterity. Each subsection of this chapter is chosen to highlight technical and non-technical topics that should be integral to mechatronics education long into the future.

The author was an invited panel member in the vigorous discussion at the Mechatronics 2014 conference held in Karlstad, Sweden in June 2014. He has worked in the manufacturing systems integration industry and academia for almost fifty years. The views expressed in this chapter are his alone, and designed to provoke discussion and hopefully bring about real advances in mechatronics education among teaching staff and administrators in the institutions of its readers. As educational delivery mechanisms migrate from the traditional lecture-recitation

D. Russell (✉)
Penn State Great Valley, Malvern, USA
e-mail: drussell@psu.edu

P. Hehenberger and D. Bradley (eds.), *Mechatronic Futures*,
DOI 10.1007/978-3-319-32156-1_15

classroom in favour of more outcome-based syllabi and technology-enhanced learning, it is hoped that the reader will be able to decide upon the best course of action to take for mechatronics and similar discipline courses of study.

15.2 The Educational Experience and Employment

Taking a rapid global scan of the educational process, it is obvious that there is no real *body of knowledge* of mechatronics as opposed to say the medical profession. It is not within the purview of this chapter to compare countries with countries, universities with universities or even precollege common core education. The objective is to highlight how differently somewhat similar materials can be taught to the students who constitute the future engineering cohort yielding to the inevitability of on-line delivery.

While on-line instruction, at the time of writing, may be in an ascendancy, the *Center for Teaching and Learning* at the University of North Carolina Charlotte (UNCC) [1] lists 150 different teaching methods, admittedly not all of which apply to mechatronics. These range from the well-known "*lecture by the teacher*" which appears as #1, to "*small group brainstorming*" listed as #150. Buried as #106 is "*the use of technology and instructional resources*." At the risk of being facetious, the chapter author's favourite is #127 "*visit an ethnic restaurant*." But, what is best for the student?

There are many instructional methods. Table 15.1 is based on a College of Southern Nevada (CSN) website [2] and summarizes some of the instructional methods that can be affiliated with the various teaching styles.

Academic readers will readily associate how classes at their institutions are conducted in the main. Following the full CSN website, the interested reader may find how these methods translate to an on-line environment interesting.

15.2.1 The Institution

In the US, there are over one thousand colleges and universities that boast having an engineering school. This number is increased significantly if the number of engineering departments in Europe, China and India are added. Most schools are regulated by governing bodies (e.g. ABET in the US) as far as the curriculum is concerned, but there is no common core curriculum for the nation.

This means that what is taught at Institution A may be covered superficially or not at all in Institution B. Overseas, the problem is worse. Some engineering schools in certain countries do not pass muster outside of the country itself. By awarding *engineering* degrees such institutions promise good jobs and better lives for their graduates only to not even be considered for a good job inside or outside the country. This is not good for the student.

Table 15.1 Instructional methods and teaching styles

Method	Comments
Lecture	A flexible method which can be applied to almost any content. Although lectures can be very engaging, they put students in a passive role. Experienced staff members can interweave their real-world experiences into course materials to show the relevance of the class **Teaching Style**—*Formal Authority*
Lecture-discussions	Combines the lecture with short question periods or a series of short question periods for students **Teaching Style**—*Formal Authority*
Demonstrations	Involves students learning a process or procedure based on instructor performance. The students may be involved in the demonstration and practice **Teaching Style**—*Demonstrator*
Simulations	Simulations put learners into seemingly real situations where they can make decisions and experience the outcomes of their decisions without the risk **Teaching Style**—*Facilitator/Delegator*
Collaborative learning	Students process information and derive knowledge through discussing course-related issues and topics with each other **Teaching Style**—*Facilitator*
Cooperative learning	Small groups of students work together to solve a problem or complete a task **Teaching Style**—*Facilitator*
Case studies	This involves individuals or groups of students working together to analyze a case, which is customarily a real-life situation which has been written up to highlight problems and solutions **Teaching Style**—*Facilitator*
Role play	Students work to solve problems through adopting the different roles associated with it. Role play involves identifying, acting out, and discussing problems. With care this can be highly effective especially in the non-technical aspects of systems engineering such as human resource management **Teaching Style**—*Facilitator*
Problem based and inquiry learning	Instructors give students a problem which the student must solve by gathering data, organizing data, and attempting an explanation. Students should also analyze strategies that they used to solve the problem **Teaching Style**—*Formal Delegator*

To rectify this problem, many well regarded colleges and universities are populating on-line and residential post-graduate courses. Mechatronics, robotics and other disciplines are popular topics in what are intended to be *educational objects*.

15.2.2 The College Faculty and Staff

University teaching staff, instructors, and professors ideally are mature and have some actual industrial experience. With no real pedagogic training, they teach as they were taught, with much theory and arguably little relevance to their students interests or final occupations. Most teaching staff have had little or no formal training in teaching, classroom management, or legal and ethical matters.

US News and World Report ranks the top schools annually but this ranking generally reflects research expenditures, the number of doctoral degrees awarded if appropriate, a tally of staff who hold a terminal degree and Fellow status within their institution. The rating may include graduation and retention rates. Teaching may be prescribed for each staff member, but it is certainly held in lower regard than funded research in contract renewal matters.

Efforts such as the UK Teaching Quality Assessment (TQA) were designed to highlight and reward good teaching practice at schools and colleges in much the same way the Research Assessment Exercise (RAE) handles research. It is the responsibility of university staff to perform both research and teaching well to promote high marks in both the TQA and RAE reviews. In the US, engineering departments are subject to a periodic nationwide ABET accreditation process but only at the baccalaureate level. But, what is best for the student?

15.2.3 The College Student

In the US for example, many engineering students spend just over two years in fairly focussed programs (e.g. electrical engineering) and may select their major while in their first or second year. Concurrent with these studies, students will be exposed to ethics, legal issues and presentation. In Europe, students may enter programs already knowing their chosen field and experience four years of topical study. Some schools inject a term of work experience before their final year while others engage final year student projects.

It almost goes without saying that successful students will have good study skills and an excitement about engineering while lackadaisical students tend to do poorly and often transfer into other (self-perceived as easier) programs or institutions. It is a well-known construct that how a student learns about Science, Technology, Engineering and Maths (STEM) before college is a major indicator as to what fields of study the college-bound student selects; this varies globally as will be shown at the end of the chapter. Despite scholarships and financial aid, location, need and social status do figure as to which institutions are feasible to an applicant.

Engineering schools worldwide are somewhat selective and require four or even five years of study for a baccalaureate degree. Mechatronics is certainly taught at the baccalaureate, masters, and doctorate levels but usually championed by enthusiastic staff. Are students attracted to post-graduate degrees to help staff with research and teaching rather than industrial employ? Is this best for the student?

15.2.4 The Mechatronics Employer

Imagine now that the student has successfully managed to gain employment in a technical company that for the sake of this chapter produces or uses mechatronic systems. Such employers have a perceived need for expertise to further their product or service and have high expectations for the incoming graduate or technician.

In the legal and medical professions, novitiates must complete residencies to become certified before being allowed to practice, whereas in engineering chartered membership in an institution is considered largely optional, expensive, and irrelevant. It is common practice for new employees shadow experienced engineers until they can be assigned to projects experts in their own right. From this, the reader can deduce why projects fail, how cost overruns happen and products never quite work as anticipated by the client. What is best for the company?

15.3 Mechatronics: A Selection of Real World Vignettes

The following contains three factual real world vignettes from the chapter author's experience designed to reflect necessary topics in mechatronic education. The corporation or company names are omitted for confidentiality reasons but hopefully the reader will find the examples useful. Each subsection will briefly describe a real system and how it was designed, how a problem presented itself, the resolution of the problem, and most importantly, what educational skill enabled the mechatronics engineer to address the problem. The first case is given in much more detail than the other two to better illustrate the point.

15.3.1 An Injection Moulding Monitoring System

Overview

An injection moulding corporation is contracted with a systems engineering company to design and implement a production monitoring system for its main location that operates up to 40 high-tech moulding machines. About 35 machines run regularly on any one day producing several tens of millions small plastic parts daily. The components are packed in boxes by weight and passed on to quality control and inventory. Figure 15.1 shows a typical injection moulding (IM) factory.

Fig. 15.1 A typical US injection moulding operation (*Courtesy of the Rodon Group, Hatfield PA*)

The factory manufactures a variety of items on a job-by-job basis. A job change on any machine requires much effort in purging the previous, coloured raw materials and the necessary mould mounted, and new liquid plastic bled through the system for the next job. The mechanic may cycle the machine many times until the new part is perfect, but these test operations should not ever appear in the production count.

Summary of Requirements
Without going into further detail, the requirements of the system included the measurement of each cycle of each machine on a 24 × 7 basis, a comparison of actual performance with the factory work order, the provision of display screens throughout the factory and the periodic download of inventory data to a mainframe computer. From a data integrity standpoint, this is actually very difficult to do because not all machine cycles produce product, e.g. a technician loading a new job or clearing a mould jam.

System Design
After meeting with the industrial client several times, Fig. 15.2 emerged as the preliminary system design. The major components are fairly standard in most industrial automation setups. The programmable logic controllers (PLC) are industrial process control agents which are resistant to power outages and are available with local storage, communication capabilities and multiple input and output data ports.
Having designed the system the following hitherto unforeseen questions were posed after a more detailed system site inspection:

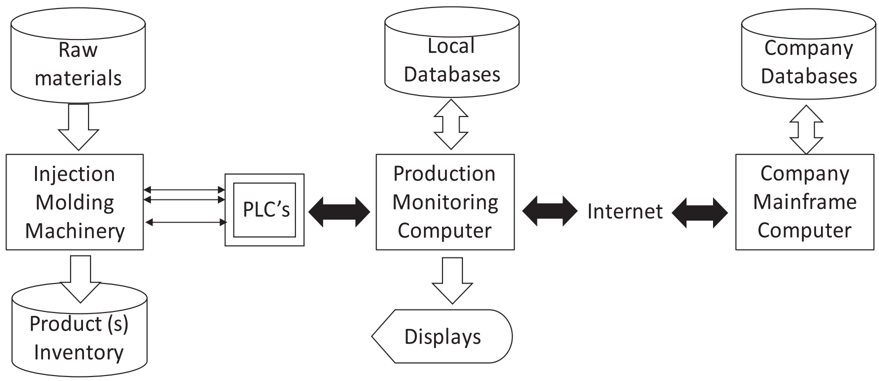


Fig. 15.2 Preliminary system design

1. How to connect machine information over long distances? The factory is over a mile long.
2. How to connect all of the system devices over such long distances? Electrical signals were all low quality with much apparent noise being generated randomly from the injection presses.
3. How much information is it useful to display?
4. How can operators and mechanics provide specific data for display?

Once these matters were resolved, which in fact did involve some redesign of the system and the purchase of additional software and hardware, the system was coded and installed.

Problem Areas
In the day to day operation of the system, the following unexpected situations arose:

1. What appeared to be random data freezing anytime during operation.
2. Data loss after a blackout or brownout of primary factory power.
3. Handling machine maintenance and repair status cycles.
4. Shift reports show incorrect times.

These problems seemed to indicate fatal flaws in the system, yet were solvable using mechatronic principles. The chapter author's solutions are summarized in Sect. 15.5.1.

15.3.2 Executing Mainframe Code on a Minicomputer

Overview
A company was using a mainframe computer for advanced CADCAM and graphics. Each design station cost over $50,000 and the mainframe lease and operating

system was over $100,000 per month. The consultant found a company that had found a way to run instructions from the mainframe on a $20 K minicomputer by making some minor adjustments to the motherboard of the minicomputer.

Overview of Invention

Figure 15.3 illustrates how the mainframe instructions were accessed and executed by the minicomputer by modification of the minicomputer motherboard with proprietary firmware. The schematic blocks shown dashed were the only firmware modifications needed. The minicomputer word size must be comparable with the mainframe instruction chip set (32 bit) which was purchased from the manufacturer.

Problem Area

The system functioned very well and the CADCAM application was successful and an inexpensive alternate to the traditional graphics workstation. One day, after a minicomputer operating system upgrade, the system completely failed to operate. Mainframe computer instructions embedded in the CADCAM sequences suddenly caused the minicomputer to return an illegal instruction trap and a complete CADCAM failure.

This problem indicated a fatal flaw in the system that eventually proved unsolvable causing the project to be discontinued. The chapter author's explanation is summarized in Sect. 15.5.2.

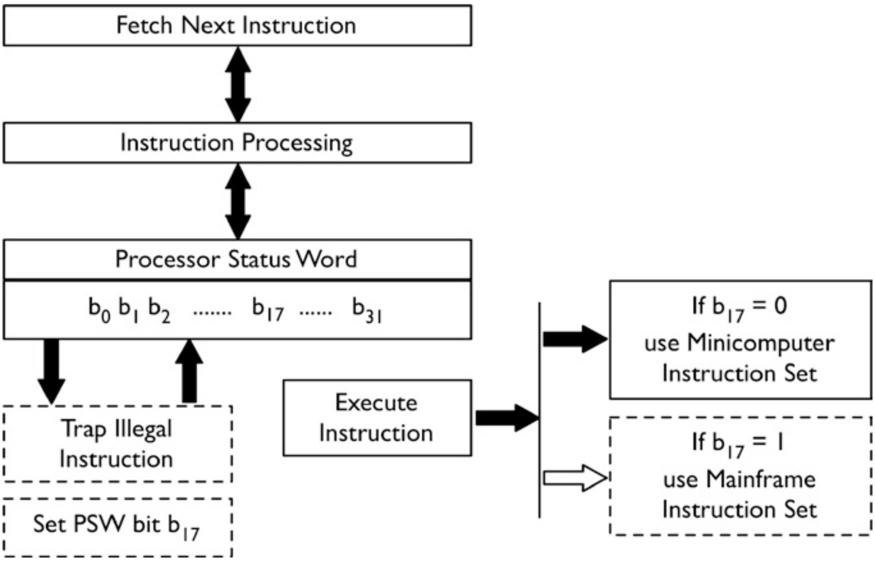


Fig. 15.3 Modified minicomputer motherboard schematic

15.3.3 A Mechanically Unstable System

Overview

Many researchers have studied various methods of inducing control into an inverted pendulum rig. This system lends itself to adaptive, intelligent, evolutionary and learning control. Figure 15.4 is a photograph of one such rig with which the author worked [3]. Essentially, the cart was driven in bang-bang LEFT/RIGHT mode on computer command. The experiment was bounded on a two meter track with crash sensors at each end. The pole on the cart was freely hinged but limited to about $\pm 10°$. If the system went out of range, the motion on the cart was stopped. The problem was to balance the pole by moving the cart left or right and should not be confused with the swing up pole balancing act.

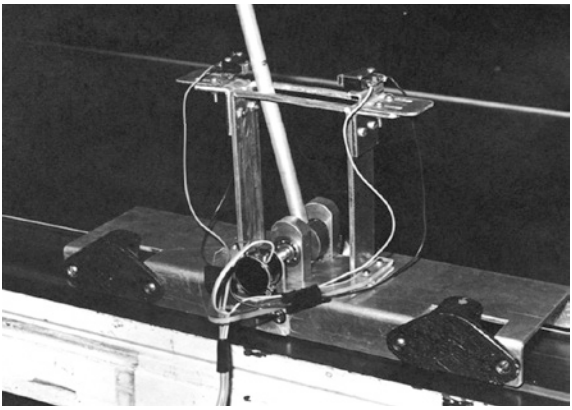
Problem Areas

The two major problems were ensuring that the system engaged its learning algorithm from an initial random but legal state so that the controller could recognize it and launch out on a control excursion, and handling slippage in the driven wheels when the cart direction was reversed. The chapter author's explanation of a solution to the first problem is summarized in Sect. 15.5.3.

15.3.4 Summary of Cases

For each of the above three cases, how these problematic situations were addressed appears below in Sect. 15.5 to encourage readers to discuss their own ideas with those of their students before reading that section. After reading the author's comments, readers should discuss then what educational modules at their institution or company would have enabled the novitiate engineer to address those problems?

Fig. 15.4 A trolley and pole experimental rig



Perhaps the missing educational experiences in our colleges and universities are in-depth coverage of systems engineering and system integration.

15.4 Systems Engineering and Systems Integration

In the cases given above in Sect. 15.3, it should be apparent that the designs of the system components, the integrated system, and even the placement of the system within its global domain (a.k.a. in a system of systems) rely heavily on the understanding of systems engineering and systems integration.

15.4.1 Systems Engineering

Perhaps the clearest definition of systems engineering is found on website [4] of the International Council on Systems Engineering (INCOSE) from where the following quotes are taken:

> (INCOSE) … represents systems engineering professionals from industry, government, and academia worldwide. It strongly believes that the fundamental principles of systems engineering have an important role in the education of all engineers, regardless of their specialty, as well as professionals who work with systems engineers but do not have an engineering background.

The same website explains the nature of the discipline and its truly outcome-based focus.

> Systems engineering is an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem.

Specifically, systems engineering is an integrative paradigm that for many years was never taught in engineering colleges, assuming that graduates of their programs will pick this up later in their careers.

> Systems engineering integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation. Systems engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs.

15.4.2 System Integration

System integration is a well-known subject in computer science and IT and has come to mean the assembly of software systems using *plug-and-play* paradigms from software located as COTS (*Code off the Shelf*), SaaS (*Software as a Service*) and of late Cloud Services. In the software realm, the major integration issues in an open architecture environment are system and application configuration. In this activity, the integrator has to skilfully slot the application into sets of code that may have been written externally in another language. Enterprise software systems such as SAP® require the use of many configuration forms and data manoeuvres before a manufacturing company can benefit from its complexity and information power. Most problems arise from hardware failures, internet issues and misfits in terminology and usage.

In mechatronic engineering systems integration problems arise from combinations of mechanical, electrical, computer and systems disciplines. Solving in one area may cause sudden failure on another front. The second Sect. (15.3.2) is an illustration of how a project failed through no fault of its own as explained in Sect. 15.5.2. It was actually the reluctance of the minicomputer vendor to make a simple revision to their operation system that caused the failure.

Computer engineering and computer science programs usually include some information integration, database and internet-enabled modules. Formal engineering programs by and large contain very little curricular coverage of system integration. Warminki and Ikonomov [5] opine that: *"… the basic engineering curriculum fails to teach valuable skills in the areas of*:

- *Knowledge management/documentation/recall and reuse.*
- *Working in cross functional distributed teams.*
- *Critical thought in the framework of product design.*
- *Design methodology including: translation of vague requirements to engineering specifications, failure mode identification and effect analysis, total parameter and tolerance product design, manufacturing execution, function as a member of a team to undertake the analysis and integration of automated manufacturing processes."*

This issue at their institution is being addressed by a detailed hands-on project in which students are posed with real problems to solve. In a group situation, students can engage in problem solving activities such as Scrum [6] and other similar team oriented project work.

15.4.3 Hands-on Versus Knowledge-Based Instruction

This project-based approach introduces the controversy of the educational value of hands-on "*tinkering*" by students versus a traditional solid educational classroom instruction. The popular vogue of "learning by doing" may work well in simple classroom situations, but would it work in the cases given above in Sect. 15.3? Can an impatient, paying, client be expected to wait for expertise to be learned? Section 15.2.4 is understated.

Formal engineering programs especially those under accreditation control are loath to forfeit more classical topics in favour of mechatronics or systems engineering. Many schools have introduced one or two year taught master's programs in mechatronics. These are more popular in the US than in Europe. In all, it is the experience, enthusiasm and focus of university staff that are charged with the trust of producing ethical, worldly wise competent engineers of all disciplines. Much project work is undertaken on an individual level with little interface with other students, whereas in industry the ability to work in a team is a much sort after skill.

15.5 Solutions and Educational Sources to Case Issues

The following are outlines of how the problematic areas of each case were resolved, but readers may want to discuss other solutions with their colleagues and classes. Much more detail is given to the first case to illustrate the complexity of mechatronic systems and because it was housed in a real-world industrial environment. The second focussed on the need for a fairly deep knowledge of operating systems and firmware, and the third on mechanical design and the use of timed software.

15.5.1 An Injection Moulding Monitoring System (Case 15.3.1)

The solutions to the problems introduced in Sect. 15.3.1 are summarized below but it should be clearly understood that this is not an exhaustive list.

Problem (a) and (b)
These questions focussed on the long distances connecting devices and the low quality and high noise electrical signals.

Solution—The use of shorthaul modems and a check on all wire shielding in the factory roof helped with this problem. A better, if more expensive, solution would have been to rewire using fibre optic cables.

Educational Objects—The engineer needed to be conversant with modems, communications and fibre wire connections over long distances.

Problems (c) and (d)
These introduced the issue of good data collection, displays and factory floor inputs.

Solution—It is essential that a focus group that includes the industrial client and factory floor personnel decide what data is to displayed on the shop floor. It became apparent in the system in question that shop floor data needed to be collected from the operators. This data then identified the need and nature of a machine breakdown, etc. It was necessary to instal microterminals and integrate this data into the database using data fusion techniques.

Educational Objects—The system designers needed a deep understanding of database design and fusion, and human computer interaction.

Problems (e), (f), (g) and (h)

These all occur during the operational phase of the system from time to time. In the original system, the data collection and all database operations would freeze mimicking the effects of a power outage.

Solution—The design and implementation of the factory software required a level of system intelligence so that temporary problems and failures could be detected and "self healed" to avoid loss of data. The actual system included programmable logic controllers (PLC) in which front end intelligence was embedded to temporarily store data during a system pause or stoppage.

Educational Objects—The mechatronics engineer needed to understand file locking and system programming to free locked files and folders. Real time operating system design knowledge is essential as was a familiarity with available industrial components.

15.5.2 Executing Mainframe Code on a Minicomputer System Failure

How the system works

Figure 15.2 depicts how the proprietary firmware purchased for modification of a minicomputer motherboard utilizes an unused bit 17 in the minicomputer's 32 bit processor status word (PSW). The operating system kernel allowed system users to access all PSW bits in high priority tasks. Included in the PSW is a bit 3 that traps an instruction error. It was this bit (bit 3) that is set when the minicomputer attempted to execute a mainframe instruction. If the executive program detects such an event, it sets what was the last unused bit (bit 17) that was designed to direct execution to the additional hardware for execution.

Reason for failure

The minicomputer vendor issued an update to the operating system that innocently used that bit (17) for a new elaborate print function. The operating system software team had spent many hours developing this new function that would benefit all of its other customers. The CAD/CAM project was cancelled.

Educational Objects—For the mechatronic engineer to detect, this would require a fairly high level of computer architecture, systems programming, and firmware. As an aside, advanced negotiating skills might have saved the project!

15.5.3 A Mechanically Unstable System

Randomized but Legal Initial State System

Many pole and cart systems begin with the pole being held vertically near the centre of the track. Upon release, the system is engaged and the process proceeds but always from nearly the same initial state variable values. This is a real flaw in the system. In the case in question, in order for the trolley and pole logic to engage its learning paradigm from a random but recognizable initial state, it was necessary to construct a startup subsystem that drove the cart in one direction for some random time and then reversed the cart direction for a shorter random time and then reversed it again. This would jerk the pole from its initial steady state resting position into a dynamic state but will not allow it to gain enough momentum to fail. During the startup process the control system monitored the state variables When the starting system entered a state in which the system's bang-bang controller value coincided with the startup value, the startup logic was disconnected in favour of the system.

There many other such examples where readers may choose to insert their own examples from their own experience using this approach.

15.6 Conclusion: A Global Problem with Local Solutions

Addressing future educational methods "*The answer is not to be found by looking in the rear view mirror*" So states Marshall McLuhan quoted in a recent *Educause* article by Brown [7]. Brown discusses concepts such as Adaptive Learning Technologies, Learning Spaces, Learning Analytics, and Next Generation Learning Management Systems and focuses on how students must navigate a way through a pathway or *swirl* of instructional experiences.

This certainly has elements of truth but might be an oversimplification. Engaged faculty who are able to bring their research or other technical interests into the classroom can not only hold the attention of their class, but also create a learning environment that causes students to be life-long learners, ethical, and innovative. Looking again at Sect. 15.5 where plausible (and actual) solutions are listed, readers should consider where these skills are being taught at their own institutions.

This matter is not limited to North America or Europe, but is a global malaise in what are often classified as *good* institutions in China, India, Singapore, Australia and many other countries.

A ready solution might be a better understanding and use of continuous professional education (CPE) modules such as offered by universities and the professional institutions such as the IMechE, IET, IEEE, ASME and the like. Such programs can help retrain more senior engineers as well as fill in the gaps in new hires. For a deeper coverage of mechatronics many institutions are offering taught masters programs which can be face to face or on-line. In these programs, students are already degreed engineers and therefore can focus on mechatronic issues such as described in this chapter without much mathematical or basic engineering review.

The intent of this chapter has been to introduce some concepts of how mechatronic systems posit a variety of problems for which students, even at the doctoral level, may have had no in-depth instruction and who do not yet possess the savvy of an experienced engineer. Reference to statistical data has been largely avoided as numbers change so rapidly from year to year and are provided by unreliable sources.

References

1. Teaching Methods. www.teaching.uncc.edu/learning-resources/articles-books/best-practice/instructional-methods/150-teaching-methods. Accessed 1 July 2015
2. Different Instructional Methods. www.csn.edu/pages/2359.asp. Accessed 13 July 2015
3. Russell DW, Rees SJ, Boyes JA (1977) Self organizing automata. In: Proceedings of 1977 conference information and system sciences. Johns Hopkins University, Baltimore, April 1977
4. www.incose.org/AboutSE/SEEducation. Accessed 20 Oct 2015
5. Warminski F, Ikonomov P (2007) Teaching manufacturing systems integration in educational institutions through hands-on experience. In: Proceedings of 2007 spring conference american society for engineering education north central section, West Virginia Institute of Technology (WVUTech), March, pp 30–31
6. Scrum Methodology & Agile Scrum Methodologies. www.scrummethodology.com/. Accessed 5 July 2015
7. Brown M (2015) Trajectories for digital technology in higher education. *Educause* July/August:16–27