Reconstructing Constructionism

Richard Noss • University College London, UK • r.noss/at/ioe.ac.uk
James Clayson • American University of Paris, France • jclayson/at/aup.edu

> Upshot • Constructionism must return to its epistemological roots to make any lasting impact on education. Constructionism should be transformed from a framework of action into ways to conceptualize and record what people actually do in constructionist environments so that theories of knowledge-building acts can be tested and the designing of those environments can be made more effective.

It is now a quarter of a century since the idea of *constructionism* was launched by Seymour Papert – the *n*-word rather than the *v*-word, constructivism (Papert & Harel 1991: 1). While the latter idea captured nicely the psychological substrate on which *all* learning (irrespective of teaching) is built, the *n*-word sought to develop a theory of pedagogy that could foster learning. More than that, while the constructionist project seems like a pedagogical theory, it is as much a theory of epistemology as one of pedagogy: understanding the development of the structure of knowledge is part of and integral to the encouragement of an inclusive and powerful pedagogic theory and practice.

Constructionism symbolised a way of thinking about learning, a metaphor for the ways that human beings come to learn most effectively: building a model, reflecting on it, debugging and sharing. All this was to be achieved in ways that were "reasonably straightforward" thanks to digital technologies – in particular what Seymour called its "Protean" nature, i.e., its potential to present itself with any number of faces. The Logo programming language (Papert 1980) was emblematic of this approach and has left its mark on the educational world, most recently the advent of Scratch with its 10 million projects and which has already shown signs of contributing to the creation of a broader constructionist methodology (Brennan & Resnick 2013).

Despite this success and the current rebirth of the programming idea in many countries, there is little sense yet of fundamental change in the practices of teaching and learning. There are several reasons for this, including overtly socio-political aspects: not all educational endeavour seeks to create independent activity and foster independent thought. And we have yet to build a literature that provides the nuance and richness to constructionism that would be necessary for it to inform education in the large.

Another reason for constructionism's slow burn is that constructionism is often confused with its psychological cousin, constructivism – a word that has all but lost its actual meaning in the rush to embrace an alternative to behaviourism and its offshoots; a meaning diluted to the point that almost any pedagogy is routinely described as "constructivist," as if a recognition of how humans learn is sufficient for prescribing how and what they should and could be taught.

But perhaps the fundamental difficulty of constructionism as an organising framework is simply that it is, at core, an epistemological idea, and one that has been insufficiently theorised by researchers and practitioners. There is an extreme reluctance in the educational enterprise to discuss *what* to teach in the light of new tools and new goals for the curriculum, a reluctance whose inertia pales into insignificance in the face of a simple evolution of pedagogic approach. Compared with re-evaluating what can be taught and to whom, the switch from instructionism to some other ism that recognises the complexity and heterogeneity of learners is far from unproblematic. The problem is that if conceived merely as a pedagogic strategy, constructionism does not offer in concrete terms much more than a host of other worthy slogans such as "discovery learning," "exploratory learning" or "enquiry based" learning.

Of course, even if there were to be a widespread wish to change what to teach, in the interests of engagement and accessibility for example, we should not underestimate how hard the endeavour is.[1] Where does one start and how does one progress when every design decision has enormous consequences for possible actions, the language in which the actions are expressed in a model, the language through which the actions are shared, the planned sequence of work, etc. All of the above can only be undertaken in an iterative spiral of "design-and-test" among an interdisciplinary team of educationists, computer scientists and teachers.

What kind of a theory is constructionism?

Our starting point is a seminal paper by Andy diSessa and Paul Cobb (2004). In it, they argue for the importance of theory in educational design experiments, and they survey different roles played by theory in design. They differentiate between four types of theory – from "grand" theories such as Jean Piaget's constructivism (which they properly point out was not intended to and largely fails to inform design) to "domain specific instructional theories," which involve testable conjectures about learning processes and how to devise pedagogic situations that foster them.

1| At a conference around mathematics education, Papert set us the challenge to change just 10% of the curriculum in the light of our discussions around the use of technology. We fear we did not meet this target. Some results were achieved, as recorded in Hoyles & Lagrange (2009).

http://www.univie.ac.at/constructivism/journal/10/3/285.noss

DiSessa and Cobb suggest that constructionism, like "learning by designing" falls into a category they call "Frameworks for Action," and they argue that while these frameworks do provide some heuristic power and structure to the design of learning environments, they typically…

> "do not cleanly separate their scientific claims and validation from their suggested actions. That is, the theory or theories behind frameworks for action are relatively inexplicit, complex, and often involve multiple very diverse elements that cannot plausibly be brought under a single umbrella." (diSessa and Cobb 2004: 82)

DiSessa and Cobb argue for the need to "manage the gap": the failure of most frameworks to accommodate the complexity and interactions between the elements of instruction. It is trivial to note that instructional effectiveness depends on many variables, not least the nature of technology, a field that is chaotic in the literal sense: tiny changes in, for example, the user interface can make massive changes in learning. The primary point is that in order to "test" theory, it is necessary to maintain a gap between the pedagogical strategies at stake and the theories that motivate them: one cannot prejudge what is to be found when attempting to "apply theory to practice."

The point is that in designing learning environments that integrate digital technologies, one needs to recognise that the tools made available shape the activity in ways that to some extent are predictable but in some are not. In addition to considering the specific affordances and constraints of different digital technologies for structuring learning experiences (including various software packages, hardware configurations and the Internet), there are also implications of design decisions on tools, curriculum, teaching and learning and, of course, assessment – all huge issues in education.

DiSessa and Cobb claim that it is necessary to "develop theoretical constructs that empower us to see order, pattern, and regularity" in the settings under investigation. Their research argues that students' construction can be studied under proper conditions. However, designing those conditions in a variety of different disciplines – not just mathematics and science – is the crux of the difficulty confronting constructionism as a theory.

The prevalent assumption underpinning most educational research – and by implication the producers and consumers of education research – is that the fundamental concepts remain invariant over time and technologies. It is tempting to take this observation as merely trivial: educational change is slow, it seldom takes account of the possibilities of knowledge transformation, and it is almost always concerned only with teaching more effectively, rather than learning within new epistemologies.

While this is correct, it misses a key point about constructionism. When we build, we build with things – not just ideas. Of course, if we design properly, the things we build with have an epistemic foundation – of "powerful ideas," say, that students are supposed to bump into, or perhaps we should say, create. But the ways the things are connected in construction, the relationships between them, and the behaviours they are given have to be expressed in the system of the things, not in the system of ideas. We *could* express the fact that the paragraph settings of this paragraph are contained in the final paragraph marker as a line or two of code; but as we are building this paragraph, it is much more natural to say (to ourselves), "If we merge this paragraph with this one, it will inherit the second one's properties. Note the informality of our expression: 'this' means nothing outside the situation of writing."

This particular property of construction systems (such as programming languages) is both a powerful advantage and a difficulty. It is powerful because the complexity of an idea often inheres in the way it is represented, but that representation is not just in the program code. The act of programming encourages a form of layered model building of text, diagrams, group narratives and sketches that surround and embellish the code. We use these tools / objects / words / phrases to speak freshly. In a way, we build a new language of specific objects by using a specific source language. How to extend that new vision into other specifics is tricky.

Let us give an example of this problem. Over many years we noticed a recurrent pattern in students building computational expression for mathematical and scientific ideas. We saw that while they seemed often clearly able to abstract from the particularities of the activity, as evidenced by an often implicit recognition of the relationships between variables, these abstractions did not resemble in their expression the standard forms of algebraic or even quasi-algebraic representations. Naturally enough, they employed the tools – objects and relationship between objects – that they had used successfully in the activities. The tools-in-situation, in other words, acted as a means to express abstractions that might not have been expressible in standard forms: we called these "situated abstractions" (Noss & Hoyles 1996) to try to capture this idea.

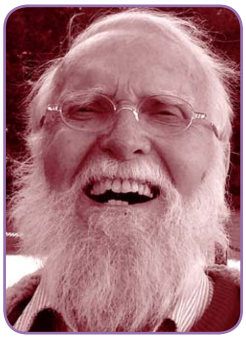
Situated abstraction identifies and organises a class of behaviour and expression that occurs in the context of activity in constructionist environments. Like any useful theoretical idea, its power lies in its application – in the potential of the idea that started as an observation of behaviour to influence and shape behaviour.

This is a major challenge of research: to transform constructionism from a framework for action into a set of ways of conceptualising what people do in constructionist environments that can simultaneously assist in designing those environments. We attempt to take a step in this direction, by seeking to make explicit what we see as the defining characteristics of a constructionist agenda, which together define its distinctive character.

Characteristics of a constructionist agenda

Central to any notion of constructionism, and its first defining characteristic, must of course be the idea of *modelling*; that is, by creating external building blocks by a process of building, reflecting and debugging, learners can develop relevant internal knowledge structures. Modelling, approached in this way, promotes the learning of *powerful ideas through use,* in contrast to the conventional way of much teaching (Papert's 1996 "Power Principle"). A key rationale for modelling (at least in the context of mathematics) involves *using* and *discriminating* crucial invariants, *generalising* and

RICHARD NOSS
is director of the London Knowledge Lab, an interdisciplinary centre of the Institute of Education within University College London. He is Professor of Mathematics Education at the UCL-IOE, holding a master's degree in pure mathematics and a PhD in mathematical education. He was co-founder and deputy scientific manager of Kaleidoscope, the European network of excellence for technology enhanced learning, and was the director of the UK's Technology Enhanced Learning Research Programme, funded jointly by the ESRC and EPSRC. His research has focused on the design of constructionist computational environments for learning a range of ideas – mostly mathematics-related. He has extensively researched the kinds of knowledge needed by employees in technology-rich workplaces, and appropriate ways to harness technology to foster this knowledge.



JAMES CLAYSON
is Professor Emeritus at the American University of Paris (AUP) where he taught applied mathematics and visual thinking for thirty years. He specialized in designing computational environments where liberal arts undergraduates could explore the power of building personal models that link the visual, the qualitative and the quantitative. He won AUP's outstanding teacher award three times. His early book, Visual Modeling with Logo: a structured approach to seeing (1988), was published by MIT Press in its series of constructionist explorations using the Logo language. His latest book, A Computational Eye, (in process) uses the Python computer language and was tested on students at Deep Springs College (California) where he was a visiting professor in 2015. Jim was educated at MIT, the University of Chicago and the School of Oriental and African Studies (University of London).

synthesising, within a framework of iterative design, a necessary condition if learners are to develop *agency* over the evolving knowledge. Thus modelling emphasises the utility of a mathematical concept from the learner's perspective. Jere Confrey and Alan Maloney describe this process in a similar vein:

> "[T]he modeling produces an outcome – a model – which is a description or a representation of the situation, drawn from the mathematical disciplines, in relation to the person's experience, which itself has changed through the modelling process." (Confrey & Maloney 2007: 60)

Students live in a world increasingly permeated by technology – the internet, satellite communications and mobile phones. Their lives are managed by numerous technological systems, many of which are largely invisible (transportation, finance and loans, manufacturing, demographics, medicine, and so on; see, for example, Hoyles et al. 2010). So, a second characteristic of the constructionist agenda should embrace the issue of *accessibility* to the modelling process: learners develop an awareness of the existence of models and how they shape actions; then they are provided with a glimpse of how this happens.

Our approach to this challenge has been through *layering* of mathematical and scientific principles and abstraction, and thus embedding increasing problem-solving complexity into the software. This again affords a higher degree of user agency: learners can decide how deep they dig into the "why" of the software feedback, or if they want to be able to edit or extend the models. The idea of layering is a third component of the constructionist agenda.

A fourth characteristic is *tapping into youth culture*, or more generally, seriously seeking to engage with learners' agendas. This effort should not be underestimated and is especially important for subjects such as science and mathematics, which carry considerable social capital, yet are easy for students to dismiss as irrelevant, boring and hard in a world of digital images, animations, instant information and communication. We have tried over many years to design and build engaging environments, in which the knowledge (say, some mathematical content) is actually *needed* for students to achieve the goals of learning.

This brings us to the fifth characteristic: that the knowledge is made visible by being *represented* in a language with which learners can express themselves. Just what "language" means in this context is crucial, and we will not explore this complex issue further here.

The sixth characteristic moves the focus of attention in the constructionist agenda more towards appropriate pedagogy: effective student learning is promoted through long-term engagement in collaborative projects during which students take individual and collective responsibility (e.g., Harel & Papert 1991) and there is *sustained* emphasis on content knowledge. This last characteristic points to *collaboration*, which is worthy of a separate defining characteristic of a constructionist agenda, not least as we are seeing rapid developments in the ways that it is possible for students to share resources and ideas and to collaborate through technological devices, both in the same physical space and at a distance.

Most of the above criteria imply considerable investment in *design*, a crucial dimension in the educational use of technology, but one whose difficulty tends to be seriously underestimated. Particularly in the case of widely used educational tools, decisions taken by a small number of designers shape the way educators have to think about teaching and learning with technology. Most digital technologies do not make explicit how they work or how they could be used in education. This means that taking account of their design, particularly in terms of implications for epistemology, is a central challenge. But, as we attempt to incorporate new technological tools into teaching and learning, we must seek to make progress in trying to understand how the related epistemological structures are *mediated* by learning communities, and reciprocally, how learning communities are shaped by the artefacts and technologies in use. Bringing in tools to foster collaboration brings more complexity to the issue of design, since again the technical aspects shape what students can do with the technology, what they can share and how they can interact.

This means that we are designing for a moving target in all these directions: a challenge for designers, researchers, learners and teachers. Yet the implications for learning and teaching are beginning to be explored and appropriate theoretical frameworks put in place.

Note

An earlier version of this paper was given at Constructionism 2010, Paris, France.

References

Brennan K. & Resnick M. (2013) Imagining, creating, playing, sharing, reflecting: How online community supports young people as designers of interactive media. In: Lavigne N. & Mouza C. (eds.) Emerging technologies for the classroom: A learning sciences perspective. Springer, New York: 253–268.

Confrey J. & Maloney A. (2007) A theory of mathematical modelling in technological settings. In: Blum W., Galbraith P. L., Henn H.-W. & Niss M. (eds.) Modelling and applications in mathematics education: The 14th ICMI study. Springer, New York: 57–68.

diSessa A. A. & Cobb P. (2004) Ontological innovation and the role of theory in design experiments. Journal of the Learning Sciences 13(1): 77–103.

Harel I. & Papert S. (eds.) (1991) Constructionism. Ablex, Norwood NJ.

Hoyles C. & Noss R. (1987) Synthesising mathematical conceptions and their formalisation through the construction of a Logo-based school maths curriculum. International Journal of Mathematical Education in Science and Technology 18(4): 581–595.

Hoyles C., Noss R., Kent P. & Bakker A. (2010) Improving mathematics at work: The need for techno-mathematical literacies. Routledge, London.

Hoyles C. & Lagrange J.-B. (eds.) (2009) Mathematics education and technology: Rethinking the terrain. Springer, New York.

Noss R. & Hoyles C. (1996) Windows on mathematical meanings: Learning cultures and computers. Kluwer, Dordrecht.

Papert S. (1980) Mindstorms: Children, computers, and powerful ideas. Basic Books, New York.

Papert S. & Harel I. (1991) Constructionism: Research reports and essays, 1985–1990. Ablex, Norwood NJ.

Received: 3 July 2015
Accepted: 8 July 2015