

Teaching problem-solving using a systematic, deliverable-driven framework



Education Notes

October 2024 (Vol. 56, No. 5)

Ana Duff

PhD, Associate Teaching Professor, Ontario Tech University, educator 25+ years (elementary, secondary, postsecondary, and adult), community mobilizer and activist in Canada and beyond.

Kseniya Garaschuk

PhD, Associate Professor, University of the Fraser Valley, problem creator and problem solver.

Lynn Pacarynuk

MSc, MEd, Vice Principal at Ottawa-Carleton District School Board, secondary mathematics educator for over 20 years, lifelong reflective learner and practitioner.

Education Notes bring mathematical and educational ideas forth to the CMS readership in a manner that promotes discussion of relevant topics including research, activities, issues, and noteworthy news items. Comments, suggestions, and submissions are welcome.

John Grant McLoughlin, University of New Brunswick (johngm@unb.ca)
Kseniya Garaschuk, University of the Fraser Valley (kseniya.garaschuk@ufv.ca)

The problem of solving a problem

Ana started thinking about problem-solving, really thinking about problem-solving, when, after many years of teaching she finally heard, truly heard, the exasperations she had heard up until that point a million times:

I don't know where to start!

Which was, almost surely, followed up by

I don't know where to go next!

Until then, she addressed each of those questions every time in the context-specific way: she would rewind her thought process for the specific problem at hand, employing the infamous “reverse engineering” approach. But then she started wondering — is there a pattern in her thinking? Is there, perhaps, a way to approach this in a more general, yet specific way? She was, of course, familiar with Polya’s four principles. She even remembers printing them out once and taping them to the student desks in hopes Polya’s four-step method, handy and in-your-face like that, would decrease the frequency of the questions above and increase the frequency of success her students would achieve when tackling problems. That turned out to be a fool’s errand — students seemed no wiser. Though, as is so often the case, we learn from failures as well as from successes, or maybe even more so.

So we'd like to ask you a favour: when you get to the end of this sentence, take a moment to consider the following question:

What is the first thing one should do when asked to solve a problem?

Can you share with us what your answer to this question was, before reading further? Here is a one-question, anonymous survey to do so: <https://forms.gle/W1RVAM6PqDNeSLUy8>. We are curious.

You may have answered: first understand the problem.

Or maybe you said: well, that depends on the problem.

There are many other possibilities for your answer. What we propose is that, whether you are a mathematician or not, neither of the two answers above are particularly helpful. The first one is too vague. The second one gets us nowhere because there are infinitely many problems and classifying them all is simply impossible. However, each answer above has something that is worth unpacking so that we can indeed have a concrete and helpful first step.

A review of literature shows that there is extensive discussion on problem-solving and teaching problem-solving, with deep historical roots. The research and teaching community discussion addresses different aspects related to problem-solving, from problem recognition, to problem formulation, problem-solving, solution verification, and solution communication. When it comes to the process intrinsic to solving problems, most discussion on problem-solving techniques revolves around mathematician's George Pólya's four principles: understand the problem, devise a plan, carry out the plan, look back.

In the context of teaching problem-solving, we find that Polya's principles are too general, particularly when it comes to teaching novice problem-solvers. To a novice problem-solver, Polya's instructions for considerations in understanding the problem can feel overwhelming. Moreover, they carry great risk of information overload and derailing. In addition, we believe that they are too ambitious, particularly in the instruction to first devise a plan, then to execute the plan. In this article, we will describe an attempt to unpack Polya's general instruction in a way that guides novice (and experienced) problem-solvers in answering two quintessential questions in problem-solving:

How do I know where to start?

How do I know where to go next?

We will provide an overview of the problem-solving framework proposed in the [Problem-Solving Framework discussion paper](#) by one of the authors that works to address these questions, and more. The framework is characterized by the focus on the deliverable and the principle of just-in-time information. We will also discuss how this approach may tackle the shortcomings of current strategies and practices in teaching problem-solving in secondary and post-secondary classrooms, particularly in mathematics. In full transparency, we will also discuss some challenges that arise in applying this framework, both intrinsic and extrinsic.

We adopt from the Problem-Solving Framework discussion paper the following definitions:

- The term *problem* will mean the totality of the problem description, which consists of explicit or implicit statements of one or more deliverables, conditions imposed on the deliverable(s), and background information for the purpose of context-setting.
- The term *context* will describe the situation, background or environment where the problem is situated or specific information adjacent to a word or expression and describing its meaning, and can refer to *general context* or *specific context*, depending on the level of detail.
- The term *deliverable* means a tangible or intangible product produced as a result of a project and intended to be delivered to a customer; in our context it shall mean the specific product to be delivered to the problem-poser as part of the project of solving a problem.

Deliverable-driven problem-solving framework

How do I know where to start?

The first step should establish the target or what we refer to as *deliverable*. This consists of two components: summarizing the general context, then identifying the deliverable(s), in that order.

For example, the problem statement may be something like this:

Bla blab la blabl abla ... describe the plant... blab la blabla.

The task to *describe the plant* gives us the deliverable, which is the *description of the plant*. However, without the general context, we don't know if *the plant* refers to a sunflower or a factory or something else that can also be referred to as *a plant*. Thus, that first step of getting the sense of the general context is important to infer meaning to the words used in identification of the deliverable. It is not, however, important to know at this point (or perhaps at all) that the sunflower is 15 cm tall or that the plant is the GM factory in Oshawa.

Which brings us to the importance of the principle of *just-in-time-information*, taking us to the second question...

How do I know where to go next?

Still not by considering and listing out all the information provided!

The answer to that question will depend on what we know about the deliverable. The emphasis here is on what *we know*, and not what *we're told*.

For this, our next step is to write down or in some other way plant our deliverable statement, say:

Deliverable: description of the plant?

This we should do for three reasons: initial target-setting, easy reminder of our target when we're knee-deep in working towards it, and verification at the end that we indeed delivered what we were supposed to deliver (and did not go off on a tangent somewhere, delivering something unrelated). The deliverable statement should always be the first thing we make a note of in the write-up (or talk-up) of our solution.

We then have to exercise again the discipline of the mind by stepping away from the problem statement and instead by recalling all we know in general about the deliverable. This will include the meaning of the words in the deliverable statement within the general context of the problem, and perhaps formal definitions and the associated characteristics of the deliverable, its constituent components, and general relationships (in math, read: formulas, equations or inequalities) between the deliverable and other concepts related to it within the problem's general context. With that, our target has meaning.

Alan H. Schoenfeld speaks of this in *Mathematical Problem Solving*^[1] as "resources", when saying that "To understand why an attempt to solve a problem evolves the way it does, we need to know first what `tools` the problem solver starts with. Ideally this first category, resources, provides that kind of information. It is intended as an inventory of all the facts, procedures, and skills – in short, the mathematical knowledge – that the individual is capable of bringing to bear on a particular problem. The idea is to characterize what might be called the problem solver's "initial search space." What avenues are open, at least potentially, for exploration?"

Thus, after identifying the deliverable, we should then make a note of the general characteristics and relationships we are able to recall about the deliverable as a concept, preferably in writing or in some other way, synthesizing them when possible in a way that is descriptive or constructive of our deliverable. This will then serve to guide us as we start building it using our general knowledge and considering, eventually but not yet, the specific conditions on our deliverable provided in the problem statement. In a way, this step of "unpacking" the deliverable is what one might think of as the first step in building a pathway towards it.

Inevitably, our deliverable will be an object or a concept (we avoid here the philosophical discussion on objects and concepts) defined by a series of characteristics that may or may not be related to each other in some way, with each of those characteristics, in turn, defined through some other objects or concepts defined by ... *ad finitum* (hopefully).

For example, say that we identified

Deliverable: domain of $f(x)$?

Before we get into the specific conditions given in the problem, such as, say, the formula for , we must first unpack the meaning of the deliverable. Specifically, we must unpack the meaning of the symbols and the word domain. This is a prerequisite to the necessary conclusion that, to deliver the domain, we have to look for all possible values of that will be able to produce results using the rule that defines the function . And it is only at this point that the rule of is relevant – an example of *just-in-time-information* principle.

Yet, most of the time, when we the math teachers are presenting a solution to this type of problem (or the process of solving it), we start with the statement of the rule (i.e., the formula for) and often we do not explicitly state the deliverable and even less often do we write it down. We assume that the students have identified it and will be recalling it on their own if needed. We also often hope that students know the associated definitions and their constituent characteristics, and are able to articulate them to themselves as they are solving the problem and to others as they are presenting their solution. We often simply write the formula for and proceed with writing up inferences, and we assume that, in that moment and upon later revisits, what we *want* (i.e., the deliverable) is obvious. In other words, when we start demonstrating the solution process, by starting with the formula and following by (or completely omitting) the deliverable statement, we are sending the message to the

student that the condition on the target is more important than the target itself (or, even worse, that the condition is the only important thing in solving a problem).

Thus, to bring our discussion back to the general process, only once we made a note of the deliverable, reflected on what we know in general about it, and deconstructed it into constituent parts, we should go on a hunt for *specific* conditions given in the problem statement. In this hunt we should consider only those that are directly related to the deliverable itself or to its constituent parts. Here again we have to exercise the discipline of the mind. We must avoid taking note of conditions not directly related to the deliverable or its parts – we should make note of information only when relevant, in the moment when it is needed. We avoid to the best of our ability the attraction of “bright shiny objects” such as values and other specific descriptors that, though they may be useful later (*may* is the operative word here), they are not necessary in this moment.

In general, the answer to the question *Where do I go next?* will always, first and foremost, depend concurrently on answers to *Where do I ultimately want to go?* and *Where am I now?* The answer to the first question will depend on whether our target of the moment is the overall deliverable or, perhaps, a constituent component of our deliverable – a sub-deliverable. The answer to the second question, the placement in the *now*, will depend on what we know about our (sub)deliverable, both generally and through conditions directly related to the target of the moment.

We then use those specific conditions to further unpack the deliverable through its constituent parts, repeating the process of identification of mini-targets, or sub-deliverables. We do this until we are able to produce the deliverable within the full expanse of our general knowledge and the specific conditions expressed in the problem statement. Finally, a quick glance at our initial deliverable statement should confirm that indeed we have accomplished the task that was given to us.

The problem of teaching how to solve problems

The mess (and the fun) of it all

You are likely to agree with us that problem-solving is not, by any means, exclusively linear. It involves a messy, frustrating process that is an interchange of branching out, jumping off, and circling back, but within each of those pathway types there are portions that involve linearity: *first this, then that*. It is those components, the iddy-biddy chunks of a straight path, that give us a break, that allow us to take a breath, to get a sense of making progress, of moving forward.

It is, indeed, the mess of it, the frustration felt and conquered, what in the end gives us a sense of satisfaction and pride when we have arrived at the solution, when we *delivered*. But this mess and frustration can also be debilitating. It can prevent our students from building both the skill and the confidence in solving problems if they don't have an inkling of where to start and where to go next, then next, then next, at least occasionally and at least in a way that inches them closer to the target.

This is where we the teachers have a key role to play. We have to practice discipline on our part, the kind of discipline where we resist simply demonstrating solutions to problems and assuming that the students are sitting inside our heads, following along our thought flow. We must resist over-relying on assumptions about the knowledge of the student or the reader of the solution, most of which, in practice, are only reasonable if the student or the reader is already an expert on the subject.

Instead, we should consistently demonstrate the actual messy and frustrating process of finding a solution to a problem, but in a way that provides a string of guiding lights applicable to all problems, no matter the complexity or the topic. Summarizing our discussion above, we propose that this string of guiding lights should be the question-answer interplay of

What is the general context? What is the (sub)deliverable? What do I know in general about the (sub)deliverable? How can I use what I know to “unpack” my (sub)deliverable? What specific conditions am I given in the problem that directly relate to my (sub)deliverable? How can I use them to further “unpack” the (sub)deliverable? Is this the best I can do or is there more I can “unpack”? How can I use what I “unpacked” to “re-pack” it to get my deliverable? Did I get to my deliverable? What are the conditions on how I should present my deliverable?

At what price and for what prize?

Speaking from personal experience – this is tough!

First of all, there's only so much time. And words take time. But guidance takes words. As does the expression of our thoughts. They are so much quicker to process when they stay in our heads.

Speaking our thoughts out loud also requires us to slow them down. And that takes effort. And discipline. It's just so much easier to present the final product, a neat and clean solution, and be done with it. In a way, Brent Davies speaks of this in *The mathematics that secondary teachers (need to) know*^[1], putting forward and expounding on the notion that “the effective mathematics teacher is an expert who can think like a novice”. We extend this further to “the effective teacher of problem-solving is an expert who can *model thinking* like a novice problem-solver. Davies speaks to the danger of viewing something as “obvious” and the importance of unpacking the thing that we consider “obvious” when teaching. Using vocabulary such as “realizations”, “entailments”, and “landscapes”, Davies discusses why it is important for teachers that definitions and rules, relationships between concepts, and the contexts within the definitions, rules and relationships are recalled and processed within the given problem. What we have to own up to, however, is that this is difficult work.

Another struggle is the resistance – oof. Old habits die hard, especially when they require less effort (even at the expense of success). Starting out on a problem by writing down all of the information that is provided, without a thought of how, or even if, it is relevant requires very little mind effort. It also gives comfort – at least I have something even if I have no clue how to proceed! On the other hand, attacking a problem by starting with identification of the deliverable and digging into the brain for what we know about it requires immediate mind effort. It takes weeks, months, of true and unrelenting persistence to begin to develop in students the resistance to hooking onto bright shiny objects before securing their sights onto the target.

Yet the rewards are amazing. After twelve weeks of hard work in learning to apply the deliverable-driven, just-in-time information approach, students express genuine pride in the change to their ability to tackle problems. In math courses where this approach to teaching problem-solving has been applied, student testimonials have provided the promise of this teaching practice:

I can understand word problems better than I could before taking this course. Additionally, I am able to work much faster and understand what the question is asking for quickly.

I am most proud about learning how to properly analyze what words mean in word problems instead of focusing on numbers first.

I am better at math! I can rearrange equations properly, and have a system to tackle word problems. They don't frustrate me as much now.

I'm proud that now I can read application questions and solve them by formulating a formula through recognizing the task and walking step by step to complete that task.

I am most proud about being able to think through questions more in-depth and be able to understand them better without going straight to numbers. I am proud that I am able to understand and complete word problems easier than before this course.

Other observed but not yet specifically measured benefits of this approach include significant growth in students' skills in communicating their thought processes, their practice of inductive, deductive and abductive reasoning, and their ability to generalize processes and results beyond the specifics of a given problem. As anything in life, this approach to teaching problem-solving comes with a price, but it is one worth the prize, at least from the experience of those who practice it.

Copyright & Permissions

The Canadian Mathematical Society grants permission to individual readers of this publication to copy articles for their own personal use. Use for any other purpose is strictly prohibited. To obtain a license for anything other than copying articles for personal use, please contact the Canadian Mathematical Society to request permissions or licensing terms.

Canadian Mathematical Society — 616 Cooper St., Ottawa, ON K1R 5J2, Canada