Network models and sensor layers to design adaptive learning using educational mapping

Luwen Huang* and Karen E. Willcox†

January 15, 2021

Abstract

A network modeling approach to educational mapping leads to a scalable computational model that supports adaptive learning, intelligent tutors, intelligent teaching assistants, and data-driven continuous improvement. Current educational mapping processes are generally applied at a level of resolution that is too coarse to support adaptive learning and learning analytics systems at scale. This paper proposes a network modeling approach to structure extremely fine-grained statements of learning ability called *Micro-outcomes*, and a method to design sensors for inferring a learner's knowledge state. These sensors take the form of fine-grained assessments and trackers that collect digital analytics. The sensors are linked to Micro-outcomes as part of the network model, enabling inference and pathway analysis. One example demonstrates the modeling approach applied to two community college subjects in College Algebra and Introductory Accounting. Application examples showcase how this modeling approach provides the design foundation for an intelligent tutoring system and intelligent teaching assistant system deployed at Arapahoe Community College and Quinsigamond Community College. A second example demonstrates the modeling approach deployed in an undergraduate aerospace engineering subject at the Massachusetts Institute of Technology to support course planning and teaching improvement.

1 Introduction

Maps for education are numerous and diverse at many levels of scale. To give examples: there are degree maps that showcase paths through different majors [1], curriculum maps that trace subject sequences through a program's offerings [2], concept maps that show related topics for learners [10], and outcomes maps that support accreditation [19] and learning path generation [12, 16, 20].

*The Mapping Lab, MIT, Cambridge, MA 02139

†Oden Institute for Computational Engineering and Sciences, UT Austin, Austin, TX 78712, kwillcox@oden.utexas.edu

Scalable educational mapping via network modeling involves identifying entities and relationships amongst these entities, and representing them mathematically as a graph [19]. In contrast to a traditional table-based format, the network model explicitly represents relationships as first-class objects instead of as derived properties of other objects. This is important because relationships among elements of the model are essential to educational analytics (e.g., in pathway analyses, in understanding how content relates to learning objectives, etc.) and so the network model yields a flexible representation that enables visualization and analysis of educational data at scale. When educational maps are used for analytics and assessment, it is vital that their constituent entities and relationships are of sufficient resolution to pinpoint a learner's status and to move the learner forward. It is also vital that these maps encompass notions of sensing (i.e., inferring a learner's state) and feedback (i.e., influencing a learner's future trajectory). This paper develops a modeling framework for architecting and designing such a fine-grained sensor-enabled educational map, and illustrates its potential use as a foundational model for an intelligent tutor and intelligent teacher assistant.

In mapping a subject, entities can range from topical knowledge units to learning outcomes. Learning outcomes are statements of what a learner should be able to do; however, they are typically at a granularity level that is too coarse to support adaptive learning. Coarse-grained learning material may contain multiple sub-topics, learning activities and learning objectives, which can lead to unclear meaning in connections between learning objectives [7, 13, 18]. In contrast, adaptive learning systems and learning analytics require fine-grained learning objects [3], since in order for adaptive learning systems to correctly assess a learner's state, the knowledge units used must be granular [1, 5, 8, 15]. In this paper, we introduce the notion of fine-grained learning entities that we call Micro-outcomes. As Micro-outcomes are statements of a fine-grained skill a learner should be able to do, they will provide an effective way to infer and respond to a learner's state. Amongst Micro-outcomes there are prerequisite relationships, i.e., certain skills build on others. The idea of analyzing a knowledge domain into constituent skills and recognizing that there are prerequisite skills has long been a key idea in the concept of mastery learning [6]. Cavanagh et al. similarly break one learning objective into multiple more granular pieces that they call "learning bits" in order to design adaptive learning [4]. Here, we use network models to structure the knowledge domain and represent the prerequisite and organizational relationships amongst Micro-outcomes.

A second challenge addressed in this paper is the need for sensors that provide observational data that support inference of a learner's state. Sensors may take the form of assessment questions or digital analytics that track a learner's or instructor's actions. However, grain size is a known issue in assessment [14], and it is recognized that fine-grained statements of learning goals tied to assessments are essential to assessment design [9, 11, 17, 20]. Especially for formative use cases, it is critical that assessments should be as fine-grained as possible, ideally matching the granularity of the Micro-outcome being tested, so that precise data analytics can be collected and accurate feedback can be generated for the learner [7, 8, 15].

In this paper, we introduce a method to architect and design a network model using our high-granularity Micro-outcomes together with a sensor layer for inferring a learner's state using high-granularity assessments and digital analytics. The next section presents the theoretical framework: we begin by motivating and architecting the network model, and explain how we design Micro-outcomes. We then introduce the approach of a high-granularity assessment and/or digital tracking analytics acting as a sensor, and show how these measurements link to the network model. We apply the process of designing Micro-outcomes and assessments to a specific instance of modeling Community College subjects in College Algebra and Introductory Accounting, and describe the implementation of the resulting network model and sensors applied to an intelligent tutoring system and intelligent teaching assistant system in community college classrooms. The paper presents a second example of the approach applied to develop a network model and digital analytics sensor layer for an aerospace engineering undergraduate subject at the Massachusetts Institute of Technology.

2 Educational Mapping via a Network Model and Sensor Layer

This section first presents the network model that defines and connects fine-grained Micro-outcomes. We then describe how we architect and design a sensor layer on top of the base network model using fine-grained assessments and digital analytics.

2.1 The network model

A network model is a set of entities and relationships arranged in a graph structure in which entities are represented as vertices and relationships are represented as edges. Our previous work proposed an approach for mapping educational data with network models to obtain powerful analytical capabilities that come from making explicit the connections amongst entities in an educational system [17]. Examples of entities include: an educational institution, a department, a subject, a learning module, a learning outcome, a concept, etc.

In the network model developed in this paper, we define the notion of a Micro-outcome entity. We name a Micro-outcome for its granularity—it is a statement describing an extremely fine-grained learning outcome. Learning outcomes may be familiar to readers in education as statements of competencies; however, in this case, it is important to emphasize that Micro-outcomes are unlike common learning outcomes in this respect—Micro-outcomes are much more fine-grained. For example, Table 1 highlights a few examples of typical subject-level learning outcomes compared to our Micro-outcomes. The high granularity of a Micro-outcome in our model makes the model powerful enough to fuel many use cases, such as intelligent tutoring applications that pinpoint a user's difficulties, recommendation engines that direct students to learning resources, or evaluation tools.

Table 1: A typical learning outcome contrasted with high-granularity Micro-outcomes.

Typical Learning Outcome
Solve algebraic equations and inequalities
Micro-outcomes
Divide both sides of an inequality by a positive number
Break absolute value into two expressions
Determine if a compound inequality is a union or intersection

The network model also represents the relationships between Micro-outcomes, as well as the relationships between Micro-outcomes and other entities. Between two Micro-outcomes there may be a *has-prerequisite-of* relationship that points from one Micro-outcome to the other. This relationship represents the notion that achieving one Micro-outcome is a prerequisite to achieving the next Micro-outcome. While the notion of prerequisites is commonly used with general competencies, explicitly highlighting prerequisite relationships amongst such granular Micro-outcomes is an enabler for designing sensing and adaptive feedback strategies. Between two Micro-outcomes there may instead be an undirected *is-related-to* relationship that indicates that the Micro-outcomes are related (e.g., they relate to similar skills), but not necessarily in a prerequisite manner.

The other entities in our model are Content, Module, and Subject. A Module is a grouping of similar Micro-outcomes. This grouping is formally represented by a *has-parent-of* relationship pointing from a Micro-outcome to a Module. Similarly, a Subject is a grouping of Modules, and this grouping is also formally represented by a *has-parent-of* relationship pointing from a Module to the Subject entity. Content is related to the Micro-outcomes it addresses through addresses relationships. Figure 1 depicts the schematic of our network model with Subject, Module, Content, and Micro-outcome entities, and the relationships amongst these entities.

2.2 Architecture and design of the sensor layer

Drawing inspiration from networked systems, the sensor layer overlays the base network. The purpose of the sensor layer is to *sense* a learner's status on each node in the network as the learner traverses through the network. The sensor layer can be composed of *Assessments*, where an Assessment is a question designed to infer the learner's state relative to the Micro-outcomes targeted by

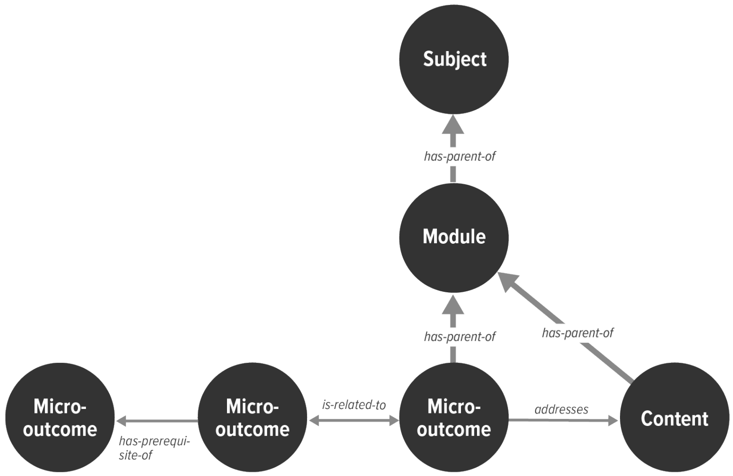

Figure 1: Schematic showing nodes (entities) and edges (relationships) in base network model.

that Assessment. The sensor layer can also include *Trackers*, which collect digital analytics about a learner's or instructor's actions (e.g., clickstream, page view counts, time on a particular screen, etc.). Figure 2 illustrates the notion of an Assessment or a Tracker serving as a sensor for a Micro-outcome.

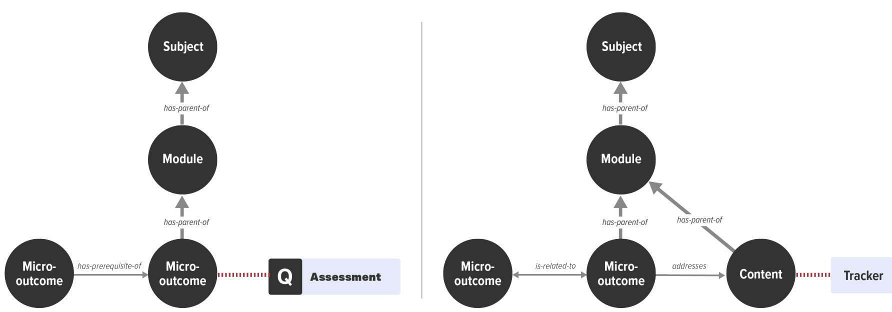

Figure 2: Assessments (left) and Trackers (right) act as sensors for inferring learner state relative to a Micro-outcome.

Trackers are code implementations designed to collect interaction information on a learner's actions, such as click interactions and time spent on a page. In the network model depicted in Figure 2, a Tracker measures actions executed on Content. Inferences about learner state leverage the underlying network model, using the *addresses* relationships that connect Content to Micro-outcomes.

Assessments can be multiple-choice or free-response, word-based or graphical, written or verbal. Because Assessments need to gather information on a learner's achievement of a Micro-outcome, an Assessment must have the same level of (high) granularity as a Micro-outcome. When a learner responds to an Assessment, the learner's response is collected as sensor data; the sensor data contains information on the learner's capability of the targeted Micro-outcome, and crucially, why the learner provided his/her response. To assess the "why" of the response, the base network model comes into play: recall that Micro-outcomes have prerequisite relationships to each other. Therefore, a gap of understanding in a prerequisite Micro-outcome is a possible reason why the learner answered incorrectly. The sensors must be designed using the base network model to enable inference of which prerequisite Micro-outcome underlies a learner's gap. This takes the form, for example, of distractor questions that target a particular prerequisite gap. Given the sensor data (learner's response) the inference of the learner's state can be based on a manually hard-coded rule, e.g., Response X always maps to (prerequisite) Micro-outcome A; it can be algorithmically-determined, e.g., an AI system can classify the response as belonging to one of the prerequisite Micro-outcomes; it can be binary, e.g., belonging to Micro-outcome A or not; or it can be probabilistic, e.g., belonging to Micro-outcome A with probability p. The existence of the base network model enables this determination. It also provides the model to determine the appropriate feedback to guide a learner through the network.

The sensor data collected provides input data to infer the learner's state relative to each Micro-outcome targeted by the Assessments. Here, another inference can be made to evaluate the learner's achievement of the Micro-outcome. The determination can be binary, i.e., "Achieved or Not Achieved"; it can be categorical, e.g., "Strongly Achieved, Moderately Achieved, Not Achieved"; it can be probabilistic, e.g., "Achieved with probability p"; or it can be mixtures of the above. Furthermore, the inference can be made with a long-memory process, in which a student's repeated attempts at a given Micro-outcome are tracked and remembered in the computation, or the inference can be made independently of previous historical data. Crucially, the base network layer joined with the sensor layer enables this inference of student state to be made at a high level of granularity. In the following sections, we demonstrate how this provides a foundation for an intelligent teacher assistant and for analytics that drive teaching improvements.

3 An Intelligent Teacher Assistant for Community College Courses in College Algebra and Introductory Accounting

This section presents the development of two specific instances of the network model and sensor layer in the mapping of community college subjects. These mappings provide a foundation for an intelligent teacher assistant system used in the Fly-by-Wire project. Fly-by-Wire was deployed at two community colleges (Arapahoe Community College in Colorado, U.S.A. and Quinsigamond Community College in Massachusetts, U.S.A.) over a period of three years, involving eight faculty members and 189 students across two subjects, College Algebra and Introductory Accounting. It is beyond the scope of this paper to detail the Fly-by-Wire project; here, we focus on the development of the network model and sensors, and how they form the basis of the intelligent feedback system.

Constructing the base network map

We map the subjects of College Algebra and Introductory Accounting as taught statewide in the Colorado Community College System. Our network model has three types of entities: Subject, Module, and Micro-outcome. Figure 3 shows the College Algebra Module "Inverse Functions" and some of its Micro-outcomes. After applying the mapping process, we obtain network models with the numbers of entities and relationships shown in Table 2.

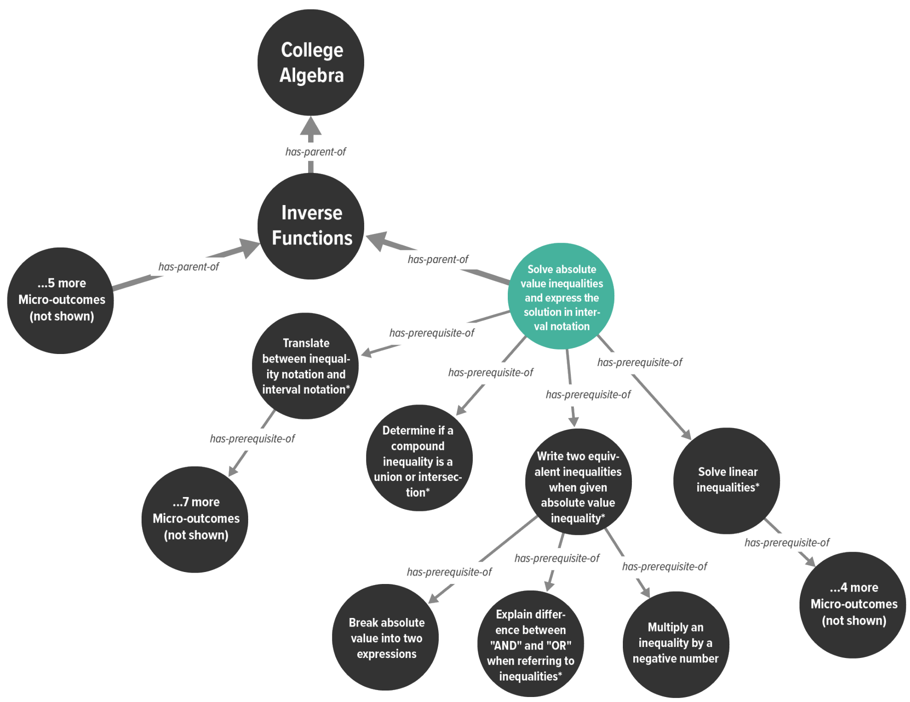

Figure 3: The "Inverse Functions" Module and some of its Micro-outcomes in College Algebra. Highlighted outcome is again shown in Figure 5. Note: most *has-parent-of* relationships to "Inverse Functions" have been omitted in the figure for clarity.

Architecting and designing the sensor layer

The next step is to design the Assessments constituting the sensor layer. To construct an Assessment, we use our network map: first, we choose a node of

Table 2: Summary dimensions of the maps and sensor layers of College Algebra and Introductory Accounting.

Entities	**College Algebra**	**Introductory Accounting**
Subject	1	1
Module	41	17
Micro-outcome	403	186
Relationships		
has-parent-of	444	203
has-prerequisite-of	446	157
Sensors		
Total number of Assessments	1091	384
Average number of Assessments per Micro-outcome	2.71	2.06

type Micro-outcome that is one of the most synthesizing Micro-outcomes, i.e., it draws from a long chain of prerequisites. Formally, this is done by computing the topological sort of the graph and identifying the nodes with the highest rank induced by outgoing edges of type *has-prerequisite-of*.

Starting with the most synthesizing Micro-outcome (with highest rank), we create a multiple-choice question designed to evaluate the learner's mastery of the Micro-outcome. We chose the multiple-choice format since students in the College Algebra course are accustomed to multiple-choice questions, but as described earlier, our framework generalizes to other types of questions. A multiple-choice question is composed of the question wording itself and the set of answer choice options. Within the set of choice options, there is one correct answer, and at least one incorrect answer. Designing the incorrect answers is key; for this we use our base network map. Using the network map, we identify the prerequisite Micro-outcomes that lead to the targeted Micro-outcome. Formally, we follow the *has-prerequisite-of relationships* to one hop away from the starting node. Given a particular prerequisite, we construct an incorrect answer that might result if the learner has not met that prerequisite. We do this for all prerequisites. Recall that there can be many different methods of determining why an incorrect response was given. In this particular instance, we deterministically assign each incorrect option to a prerequisite Micro-outcome, however, our modeling approach is generalizable to other methods of determination. Figure 4 illustrates the schematic of a multiple-choice Assessment with incorrect options that link to prerequisite Micro-outcomes. A concrete example of one such Assessment is shown in Figure 5; the top half shows the Assessment with its incorrect options (b, c and d), and the bottom half displays the Micro-outcomes that are linked to each incorrect option.

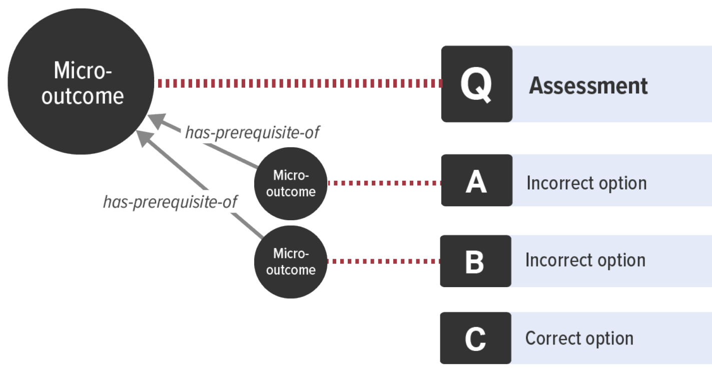

Figure 4: Schematic showing how a multiple-choice Assessment with incorrect options is linked to prerequisite Micro-outcomes

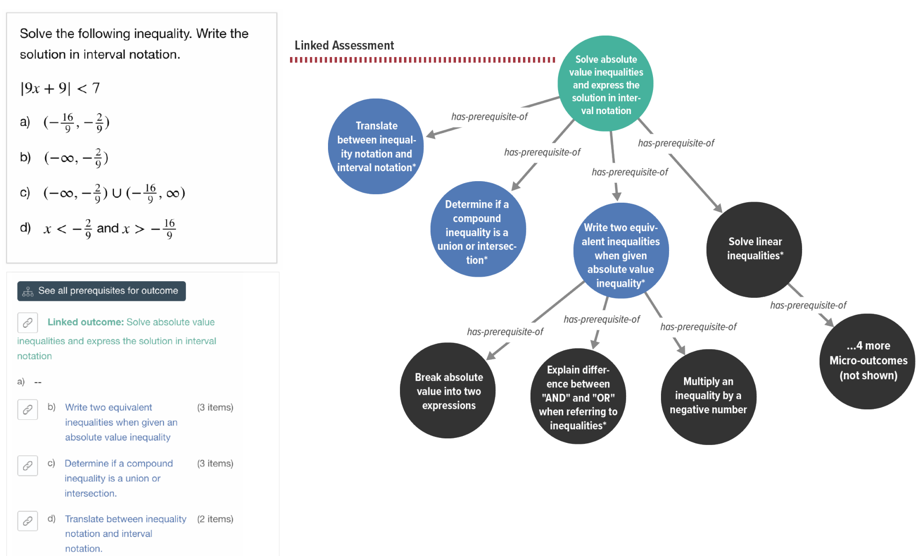

Figure 5: A screenshot from our technology of a multiple-choice Assessment for College Algebra with incorrect options (b, c and d) linked to their respective Micro-outcomes

The above describes the construction of one Assessment. To construct the next Assessment, we look to the next Micro-outcome for which to write the Assessment by traversing the graph in a breadth-first search. This yields a collection of Assessments in which there is at least one Assessment for every Micro-outcome. Table 2 summarizes the numbers of resulting Assessments for each Subject.

The base network map comprising all Micro-outcomes, Modules, and their relationships, as well as the sensor layer comprising all Assessments and their linkages, can be freely accessed at the Open Ed Graph APIs website.[1]

Deploying an intelligent tutor and teacher assistant

This network map and sensor layer form the foundations for the Fly-by-Wire Student App, an intelligent tutoring web and mobile application designed for formative assessment, and the Fly-by-Wire Instructor App, an intelligent tutoring and analytics system to help instructors identify and address areas of misunderstanding.

On the FbW Student App, students were assigned between five and seven synthesizing Micro-outcomes per homework assignment. Recall from the previous section that a synthesizing Micro-outcome is one with highest rank as computed using the base network model. For each Micro-outcome, the app displayed an Assessment targeting the given Micro-outcome. Figure 6 shows an assignment that targets the Micro-outcome "Determine the vertex of a parabola given its function and axis of symmetry." This particular Micro-outcome synthesizes six prerequisite Micro-outcomes. In the figure, the user is on the first Assessment, which corresponds to the targeted Micro-outcome.

If the student answers an Assessment incorrectly, the app presents another Assessment that addresses the Micro-outcome that is linked to the incorrect response. In this way, the student is guided in a depth-first search through the network; this results in the student most quickly getting to the most fundamental Micro-outcomes (i.e., the ones with lowest rank) that are the cause of their initial incorrect response. Here, we see a concrete instance of how an Assessment functions as a sensor, in which fine-grained data are being collected as the student interacts—the incorrect response, the time spent on a given Assessment, and any other interaction or selections the student may have with a given answer option. These fine-grained sensor data are possible only because the Assessments and their linked Micro-outcomes have correspondingly high resolution.

The Fly-by-Wire Instructor App uses the sensor data generated during student interaction on the Student App. The Instructor App highlights Micro-outcomes with which students had difficulty, and offers guidance for how to address these areas of weakness by highlighting the directed acyclic graph (DAG) formed by these Micro-outcomes and their prerequisites. For instance, consider the example shown in Figure 7: The synthesizing Micro-outcome that 11 of 22 students did not achieve was "Find all of the zeros of a polynomial function."

[1] http://mapping.mit.edu/projects/open-ed-graph/

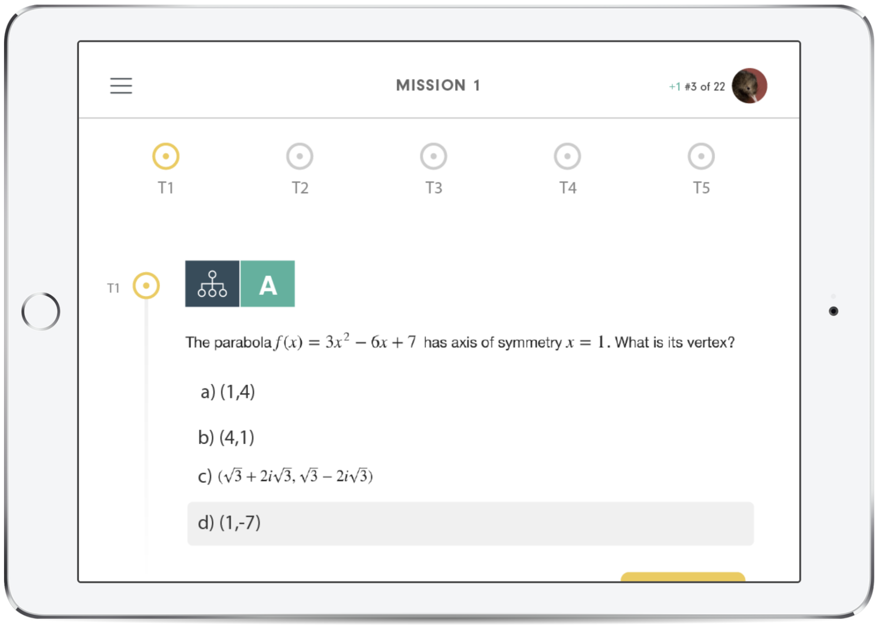

Figure 6: The Fly-by-Wire Student App delivers multiple-choice questions designed as sensors to infer student state on the network of Micro-outcomes.

The graph shown is the full DAG of the Micro-outcome and its prerequisites, and the highlighted path shows the prerequisite Micro-outcomes with which most students had difficulty. Using this network map, the instructor can then address these specific Micro-outcomes using a variety of instructional methods.

4 Fine-grained Micro-outcome Map to Support Learning Analytics in a Sophomore Engineering Subject

This section presents the development of a network model and sensor layer for the sophomore class Signals and Systems as taught in the aerospace engineering undergraduate degree program at the Massachusetts Institute of Technology in Fall 2017. In this example, digital analytics are the high-resolution sensors that track learning behavior and topical flow to assist in course planning and teaching improvement.

4.1 Constructing the base network map

The Signals and Systems subject has 36 measurable outcomes, defined by departmental curriculum planning. To construct a network model, we break these

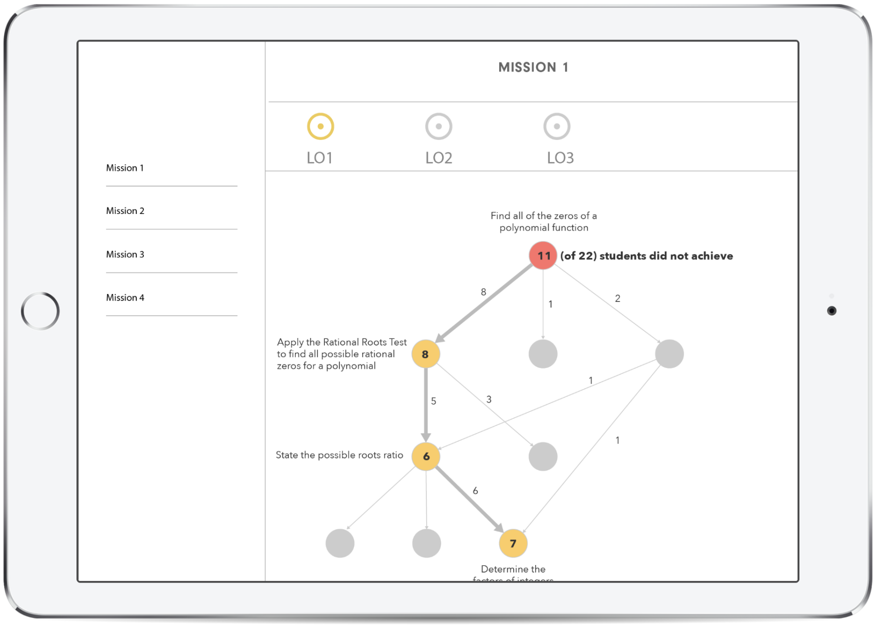

Figure 7: The Fly-by-Wire Instructor App assimilates sensor data and highlights the directed acyclic graph of the Micro-outcomes with which most students had difficulty.

measurable outcomes into 195 Micro-outcomes. We group the Micro-outcomes in 25 Modules. Each Micro-outcome is addressed by a specific section (or sections) in the lecture notes; such a section is designated as an entity of type Content. The entities in our network model are thus Subject, Module, Micro-outcome, and Content. A grouping of Micro-outcomes in a Module is represented mathematically by a *has-parent-of* relationship. Similarly, the grouping of Modules to form the Subject is represented by a *has-parent-of* relationship. The relationship between Micro-outcomes is represented by an undirected *is-related-to* relationship. The relationship between Content and Micro-outcomes is represented by an *addresses* relationship. Table 3 shows the number of entities and relationships for the MIT Signals and Systems subject. Figure 8 visualizes the Signal and Systems map with Micro-outcomes grouped into 25 Modules. [2]

4.2 Architecting and designing the sensor layer

The base network map in this application is implemented as a web application for student learning. Shown in Figure 9, the web application displays click-

[2]The interactive map can be accessed at http://mapping.mit.edu/mit-signals-systems/map-view

Table 3: Properties of the network model for the subject Signals and Systems as taught in the aerospace engineering undergraduate degree program at the Massachusetts Institute of Technology in Fall 2017.

Entities		**Relationships**	
Subject	1	*has-parent-of*	382
Module	25	*addresses*	198
Micro-outcome	195	*is-related-to*	137
Content	124		

able Micro-outcomes arranged by Module; a click to a Micro-outcome takes the learner to a Content page that addresses the specific Micro-outcome. In addition to displaying as a "list view" as shown in Figure 9, the network map is also displayed as a "map view" as shown in Figure 10. This is made possible via architecting the data backend with separation of concerns against any frontend applications.

The next step is to design the Trackers constituting the sensor layer in this application. Trackers are code implementations designed to collect interaction information on a learner, such that this information can be used downstream for learning analytics and decision-making. We attach a Tracker to every piece of Content as was shown in Figure 2, and collect the following pieces of information: the timestamp of when the learner visits the piece of Content, the location and device of the visit, the unique identifier of the learner, the time spent on page, click interactions on page, and the duration of time on page. Crucially, in addition to information collected on the current node, the Tracker also collects information on the next node, that is, the next Micro-outcome that the learner clicks to. This linked structure enables pathway analysis and inference across the entire graph. Figure 11 illustrates a single pathway undertaken by a learner in a single visiting session. Pathway analyses are valuable in helping to identify sources of student misunderstandings as well as foundational topics that relate to a large number of other Micro-outcomes. For example, in Figure 11, the Micro-outcome "Determine the Fourier series expansion of a periodic signal" is one that relates to many other Micro-outcomes in the Signals and Systems subject.

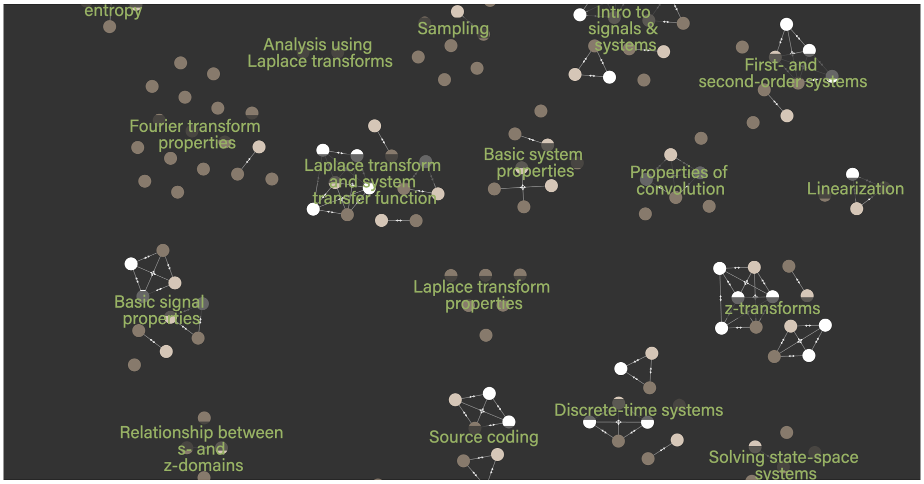

Figure 8: A visualization of the map of the subject Signals and Systems as taught in the aerospace engineering undergraduate degree program at the Massachusetts Institute of Technology in Fall 2017.

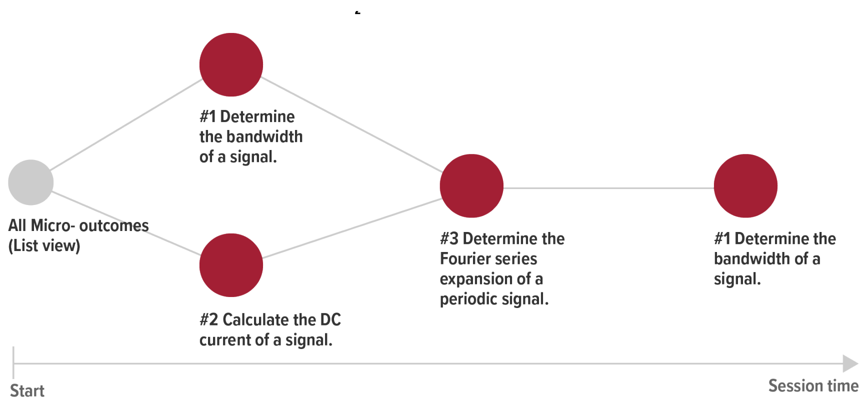

Figure 11: Interaction pathway of a single student session. The student first visits the List View, clicks to two Micro-outcomes (#1 and #2), then visits Micro-outcome #3, and finally goes back to Micro-outcome #1.

5 Conclusion

This paper has presented an approach for modeling fine-grained learning objectives (Micro-outcomes), their organizational entities, and organizational and prerequisite relationships in a network model, and then designing a sensor layer of equally fine-grained Assessments and Trackers on top of the base network map. The resulting map is a structured graph with fine-grained Assessments

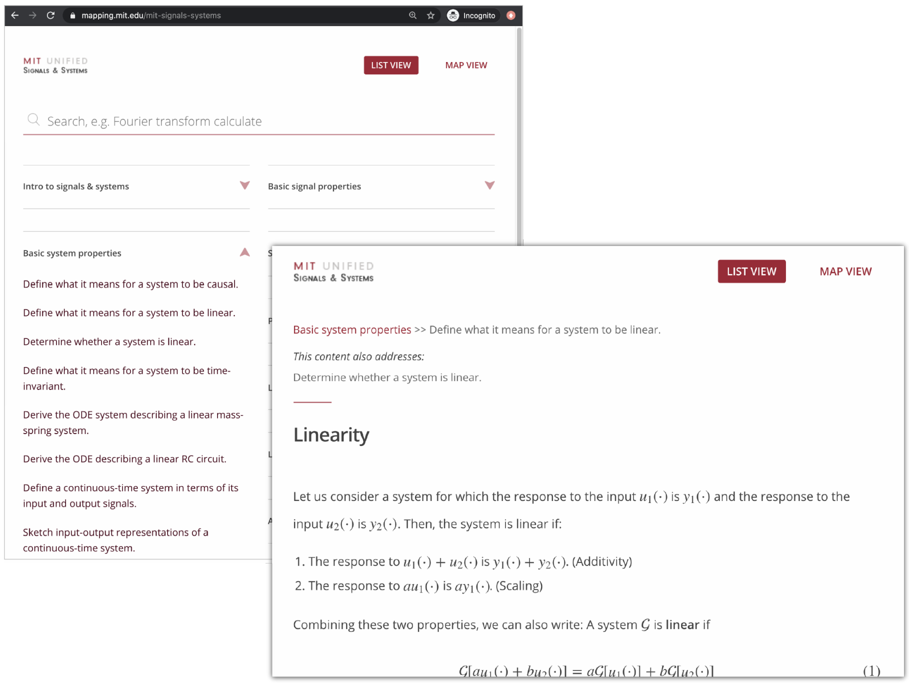

Figure 9: The map is used to create a web application that enables searching of Micro-outcomes, arranged by Module and linked to Content pages.

that provide high-fidelity sensing of a learner's state on the map. The high-resolution nature of the model enables adaptive learning systems, intelligent tutoring systems, and other forms of learning analytics. The examples presented in this paper showcase only two applications possible with the base network map and accompanying sensors. Many other applications, particularly for adaptive learning systems and learning analytics, can leverage this scalable modeling approach.

An outstanding challenge is that articulating such fine-grained statements of learning outcomes and constructing valid assessments require domain expertise and much time. However, we note that if the resulting data is stored in a technology stack that is platform-independent and is accessible via APIs, the data is easily maintained and can be scaled to many other applications. Our APIs [3] are one example of such a technology stack.

Acknowledgements

This work was funded in part by the SUTD-MIT International Design Center and the U.S. Department of Education under the First in the World program

[3]http://mapping.mit.edu/projects/open-ed-graph/

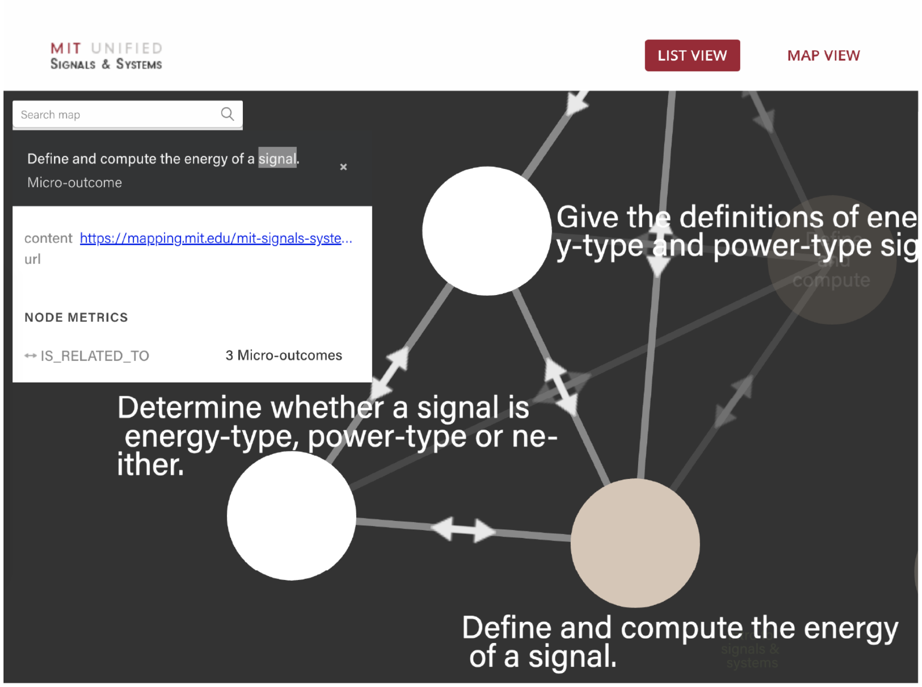

Figure 10: The network map can also be browsed in a "map view" in which nodes are Micro-outcomes, are clickable, and bring the learner to a specific piece of Content.

grant P116F150045. However, those contents do not necessarily represent the policy of the U.S. Department of Education, and you should not assume endorsement by the Federal Government.

References

[1] V. Aleven, B. M. McLaren, and K. R. Koedinger. Automated, unobtrusive, action-by-action assessment of self-regulation during learning with an intelligent tutoring system. *Educational Psychologist*, 45(4):224–233, 2010.

[2] S. Arafeh. Curriculum mapping in higher education: A case study and proposed content scope and sequence mapping tool. *Journal of Further and Higher Education*, 40(5):585–611, 2016.

[3] A. Battou, C. Mezouary, C. Cherkaoui, and D. Mammass. Towards an adaptive learning system architecture based on a granular learning object framework. *International Journal of Computer Applications*, 32(5), 2011.

[4] T. Cavanagh, B. Chen, R. A. M. Lahcen, and J. R. Paradiso. Constructing a design framework and pedagogical approach for adaptive learning in higher education: A practitioner's perspective. *International Review of Research in Open and Distributed Learning*, 21(1):173–197, 2020.

[5] J. Collins, J. Greer, and S. Huang. Adaptive assessment using granularity hierarchies and Bayesian nets. In *International Conference on Intelligent Tutoring Systems*, pages 569–577, 2005.

[6] A. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1994.

[7] C. Ellis. Broadening the scope and increasing the usefulness of learning analytics: The case for assessment analytics. *British Journal of Educational Technology*, 44(4):662–664, 2013.

[8] A. Essa. A possible future for next generation adaptive learning systems. *Smart Learning Environments*, 3(16), 2016.

[9] N. Falkner, R. Vivian, and K. Falkner. Increasing the effectiveness of automated assessment by increasing marking granularity and feedback units. In *SIGCSE '14 Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, pages 9–14, 2014.

[10] L. Fiorella and R. Mayer. Eight ways to promote generative learning. *Educational Measurement: Issues and Practice*, 28(4):717–741, 2018.

[11] S. Marion. The opportunities and challenges of a systems approach to assessment. *Educational Measurement: Issues and Practice*, 37(1):45–48, 2018.

[12] H. Miller, K. Willcox, and L. Huang. Crosslinks: Improving course connectivity using online open educational resources. *The Bridge*, 46(3):38–44, 2016.

[13] Z. A. Pardos, N. T. Heffernan, B. S. Anderson, and C. L. Heffernan. Using fine-grained skill models to fit student performance with Bayesian networks. In *Workshop in Educational Data Mining held at the Eight International Conference on Intelligent Tutoring Systems*, 2006.

[14] W. J. Popham. All about accountability / grain size: The unresolved riddle. *Educational Leadership*, 64(8):80–81, 2006.

[15] P. Reimann, M. Kickmeier-Rust, and D. Albert. Problem solving learning environments and assessment: A knowledge space theory approach. *Computers & Education*, 64:183–193, 2006.

[16] J. Seering, K. Willcox, and L. Huang. Mapping outcomes in an undergraduate aerospace engineering program. In *Proceedings of the American Society for Engineering Education 12th Annual Conference & Exposition, Seattle, WA*, June 2015.

[17] L. A. Shepard, W. R. Penuel, and J. W. Pellegrino. Using learning and motivation theories to coherently link formative assessment, grading practices, and large-scale assessment. *Educational Measurement: Issues & Practice*, 37(1):21–34, 2018.

[18] K. Thompson and F. Yonekura. Practical guidelines for learning object granularity from one higher education setting. *Interdisciplinary Journal of Knowledge and Learning Objects*, 37(1):21–34, 2018.

[19] K. E. Willcox and L. Huang. Network models for mapping educational data. *Design Science*, 3(e18), 2017.

[20] F. Yang, F. W. B. Li, and R. Lau. A fine-grained outcome-based learning path model. *IEEE Transactions on Systems, Man, and Cybernetics*, 3(2), 2017.