A Dashboard to Support Teachers During Students' Self-paced AI-Supported Problem-Solving Practice

Vincent Aleven[1(✉)], Jori Blankestijn[2], LuEttaMae Lawrence[3], Tomohiro Nagashima[1], and Niels Taatgen[2]

[1] Human-Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, USA
{aleven,tnagashi}@cs.cmu.edu

[2] Computer Science and Artificial Intelligence, Bernoulli Institute for Mathematics, University of Groningen, Groningen, The Netherlands
n.a.taatgen@rug.nl

[3] Instructional Technologies and Learning Sciences Department, Utah State University, Logan, USA

Abstract. Past research has yielded ample knowledge regarding the design of analytics-based tools for teachers and has found beneficial effects of several tools on teaching and learning. Yet there is relatively little knowledge regarding the design of tools that support teachers when a class of students uses AI-based tutoring software for self-paced learning. To address this challenge, we conducted design-based research with 20 middle school teachers to create a novel real-time dashboard, Tutti, that helps a teacher monitor a class and decide which individual students to help, based on analytics from students' tutoring software. Tutti is fully implemented and has been honed through prototyping and log replay sessions. A partial implementation was piloted in remote classrooms. Key design features are a two-screen design with (1) a class overview screen showing the status of each student as well as notifications of recent events, and (2) a deep dive screen to explore an individual student's work in detail, with both dynamic replay and an interactive annotated solution view. The project yields new insight into effective designs for a real-time analytics-based tool that may guide the design of other tools for K-12 teachers to support students in self-paced learning activities.

Keywords: Teacher dashboards · Problem-solving practice · AI-based tutoring software

1 Introduction

Much research in technology-enhanced learning has focused on creating and evaluating tools that support teachers or instructors in aspects of awareness and classroom orchestration. This work has resulted in novel tools and insight into how best to design these kinds of tools [1, 2, 4, 9, 13, 14, 16, 25]. A small number of classroom studies have

I. Hilliger et al. (Eds.): EC-TEL 2022, LNCS 13450, pp. 16–30, 2022.
https://doi.org/10.1007/978-3-031-16290-9_2

documented beneficial effects of such tools on teaching and learning [13, 16]. The current work focuses on scenarios in which students do individual, self-paced work with an intelligent tutoring system (ITS). This mode of personalized learning is increasingly common in K-12 [19, 23] and often leads to improved learning outcomes compared to instruction without this kind of software [7]. This type of software supports deliberate practice [15] in solving complex problems with "step-based tutoring" [22] and individualized mastery learning [8]. We target "real time" scenarios in which a class of students works with tutoring software, each student working individually at their own pace, and a teacher is available to help the students. The teacher monitors the class and interacts with students (often individually) to provide extra help or encouragement.

Creating teacher support tools for this kind of scenario presents several novel design challenges, compared to past work on teacher analytics tools. First, many existing real-time teacher support tools have been designed with the assumption that a class of students progresses through instructional activities in a relatively synchronized manner. By contrast, ITSs often support personalized mastery learning [8], which means that students proceed in a self-paced manner, work on different problem-solving activities at the same time, and finish milestones at different times [21]. Second, few teacher tools are designed to be used in conjunction with ITSs. These systems are typically capable of producing rich analytics [6], yet much is still unknown regarding how best to leverage these analytics to support teachers in real-time scenarios.

Recent work has started to look at these challenges by creating teacher tools for scenarios in which students ITSs (e.g., [11, 18, 25]) or other classroom scenarios [1, 16]. Some reporting tools designed for use in conjunction with an ITS support detailed monitoring of student progress [3, 5]. Other tools are helpful to teachers during classroom discussions of homework assigned through the system [14] or during lesson planning [25]. Yet other tools were designed to be independent of any learning software [2]. A few of these projects yielded implemented tools for real-time scenarios, including Lumilo, mixed-reality smart glasses that support teachers in real-time scenarios with ITSs [11]. A classroom experiment with Lumilo provides evidence that a real-time analytics tool can measurably change how teachers allocate their time and attention among students, yielding better learning outcomes for students [13]. While Lumilo provides answers to our design challenges, it requires hardware (mixed-reality devices) that is not often available (yet) in schools. Thus, how best to design tools that support teachers in helping students who are engaged in personalized, technology-enhanced, self-paced learning, is still largely an open design problem.

In the current work, we address the question: How might we design a dashboard that displays analytics from (K-12) students' work with an ITS to support teachers in aiding students in real time, during their work with the ITS? Building on the prior work with Lumilo, we conducted a process of human-centered research and design, grounding our designs in data about teachers' goals and needs, uncovered through a range of design activities. We created a new dashboard, named Tutti, within the infrastructure for development of ITSs named CTAT + Tutorshop [3].

The paper is structured as follows: After describing the instructional contexts for which Tutti is designed, we give a brief overview of the design as it is currently implemented (it is fully functioning). In the following sections, we describe the process that led to this design, present some of the insights that resulted from that process and that helped shape the design of Tutti, look at key design features and describe how they are grounded in data from our many interactions with teachers.

2 Context

The current work targets contexts in which students engage in self-paced, personalized learning with AI-based tutoring software, by now a common occurrence in K-12 in the US and elsewhere [16, 23]. It covers scenarios in which students are either present in class or work remotely, either synchronously or asynchronously. Using the tutoring software, they work through assigned problem sets, each targeting a set of knowledge components, also called “skills.” The software uses a form of AI plan recognition to assess student work at the level of problem steps and provides guidance in the form of hints and feedback. It also supports individualized mastery learning: Students move on to the next problem set only when they master the skills targeted in the current, as assessed by the system [6]. The design of the tutoring software is grounded in cognitive theory, theory of deliberate practice, and notions of scaffolding/tutoring [15].

When a class of students uses tutoring software, students typically work through assigned problem sets at their own pace. Thus, at any given point in time, different students work on different learning objectives or problem-solving tasks, even when they are working synchronously in the classroom. A teacher monitors the class and helps students in situations that the software is not designed well to handle. Other teacher goals may be to keep students on task, to keep them motivated, as well as to encourage and praise them. In remote learning, much of the communication and progress monitoring is mediated through technology. In in-person scenarios, teachers tend to move around the classroom and can talk to students to better understand their struggles or celebrate their successes. Yet it is not always easy for a teacher to assess who needs help the most, as students may hide their struggle, or, conversely, may ask for help when they do not need it urgently [24]. Further, teachers must be very efficient in their one-on-one interactions with students, as many students may need attention.

The current work builds on a proven technology infrastructure for research and development of ITS called CTAT + Tutorshop [3]. The infrastructure provides tools for building tutoring systems and for deploying and using them in classrooms. It has been used to create many ITSs [3]. It also has many affordances to support the development of analytics tools. Although our examples in the current paper tend to focus on a tutoring system for equation solving, in principle Tutti can work with any tutoring system developed within the CTAT + Tutorshop infrastructure.

3 Overview of Tutti’s Design

We briefly overview the design of Tutti in its current implementation. In a later section, we discuss key design features in more detail.

Similar to prior teacher dashboards designed for use with classes that use learning software [11, 16, 25], Tutti has a two-screen design. An *overview screen* (Fig. 1) shows information about each student in a class and is designed to draw the teacher’s attention to students who may need help (e.g., students who appear to be struggling or misusing the software) or deserve praise. At the teacher's request, a *deep dive screen* (Figs. 2, 3) shows more information about any given student. This information might help a teacher assess more fully whether communication with any given student is needed (e.g., what skills, problem types, or errors a student is struggling with).

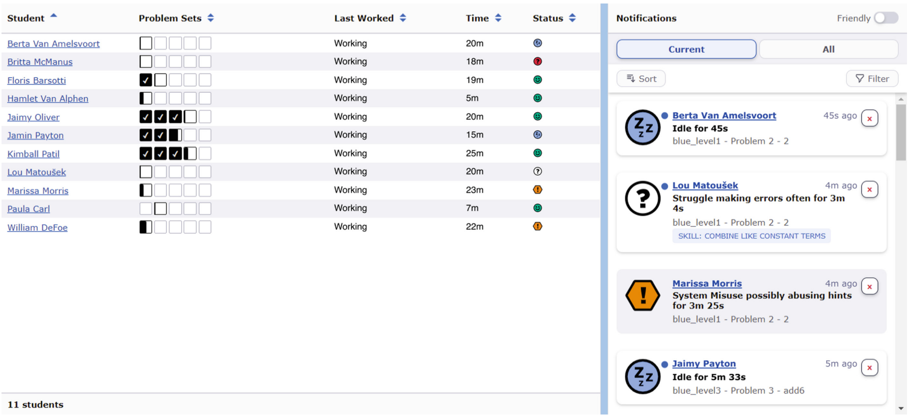

Fig. 1. Overview screen with progress table (left) and stream of notifications (right). The names are not real student names.

The *overview screen* shows a table with information about each student's progress and status (Fig. 1, left). Each of the small squares in the table represents a problem set, filled up (with black) in proportion to how far the student progressed through this problem set. A set of "indicators" capture each student's recent learning experience (shown in the "Status" column in Fig. 1). The indicators were developed and honed with frequent input from teachers in past research on the Lumilo system [11]. The indicators are: Struggle, system misuse (aka "gaming the system"), being off-task, and making good progress (so as to alert the teacher to opportunities for complimenting students). For example, to determine whether a student is struggling, their correctness rate over recent attempts is gauged, using a sliding window over student attempts. As well, the overview screen displays notifications of recent events regarding students' learning with the software (Fig. 1, right). Notifications are generated when the status of an indicator changes or when a given status has persisted for a certain threshold amount of time. For example, an idle indicator ("Zzz") appears when a student has not been working in the tutoring software for 2 min.

The *deep dive screen* (Fig. 2) provides information about a single student's progress through the assigned problem sets (top right), their mastery of the skills targeted in these problem sets (top left), and the problems they have solved (bottom left). The teacher can also look at a student's areas of struggle, defined as skills on which the student has made little progress despite ample practice, a sign that the tutoring software might not be helping the student effectively and that extra help from the teacher could be beneficial. The display of areas of struggle was highly requested by teachers. For even more detail, a teacher can look at a student's current problem solution (as a "remote peek over the shoulder," a feature also found in Lumilo [9]) or at any of a student's past problem solutions in full detail. Tutti offers two ways of doing so, both of which teachers found useful (as described below): Annotated Snapshots (Fig. 2, right panel) and Replay (Fig. 3). Both show a student's stepwise attempts at solving a problem, displayed in the tutor interface. Snapshots provide quick insight into which problem steps were challenging, as indicated by the number of errors and hints, shown with icons in Fig. 2. Replay steps through a student's attempts including errors and hints.

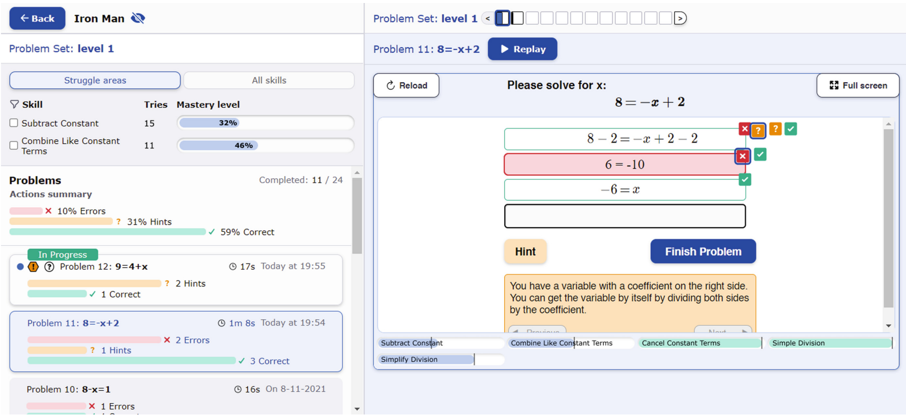

Fig. 2. Deep dive screen with information about an individual student, including areas of struggle, list of solved problems, and an annotated snapshot of a past problem solution (right)

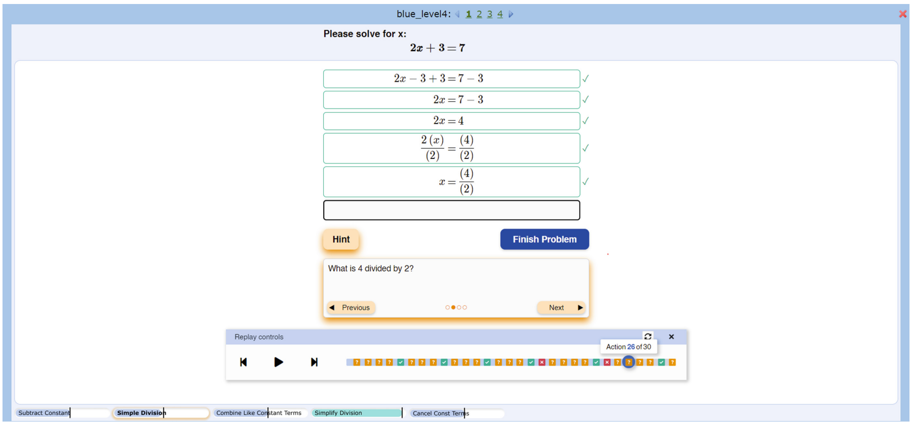

Fig. 3. Screen for replay of a student's solution to one of the tutor problems

4 Research Activities

To create Tutti we carried out a process of user-centered research and design, working with a total of 20 teachers across a range of research activities. In the current section, we describe the activities. In subsequent sections, we present the results.

1. **Discovering teachers' needs.** We conducted needs finding and concept validation exercises with six middle school math teachers from six school districts across the United States during the Fall of 2020. Three teachers were teaching fully remotely and three were teaching in a hybrid model (i.e., in-person instruction two days a week, remote instruction three days a week). We conducted six sessions, each lasting one hour per teacher. These sessions included semi-structured interviews and storyboard-based speed dating exercises [26]. Afterwards, we used affinity diagramming to reveal important themes in the teacher comments [10].
2. **Refining the understanding of teachers' needs.** We conducted speed dating sessions to solicit teachers' feedback, preferences, and motivations [26], prompted by a set of 10 storyboards. The storyboards depicted scenarios with possible dashboard designs that varied along three key dimensions: (1) Whether the instruction is in-person, remote, or a combination, (2) how examples of student work are presented in the analytics tool: as a Snapshot, as Replay, or a Live Feed of a student's screen as they are working in the tutoring software, and (3) options for teacher-student communication (audio, chat, or drawing on a shared representation of the student's problem interface combined with chat balloons). We also asked teachers what additional features and improvements they would like to see, compared to the storyboards. We clustered the resulting quotes to discover themes using affinity diagramming [10].
3. **Scoping and implementing Tutti.** Given that our needs-finding activities revealed an almost desperate need on the part of teachers to be better informed of what their students are doing during instructional sessions, we started implementing the dashboard early on during the project. We pursued the most popular ideas, including notifications of events in the learning software that might need the teacher's attention, and different ways of displaying instances of student work (Snapshots and Replay). Over time, schools gradually started shifting back to in-person instruction, which led us to prioritize features of the dashboard that were most useful for in-person instruction.
4. **Piloting an early implementation of Tutti in remote classrooms.** As the implementation effort was underway, several opportunities arose (during 2020–2021) to conduct a pilot study with an early version of the dashboard, as part of an unrelated research project. Although only the overview screen had been implemented, we figured the Tutti could still be helpful. We asked the teachers who participated in the study if they were interested in using it even though it was not yet in a perfect state. All of them agreed. We used the dashboard with three teachers in three schools in the US. One school was operating in a hybrid mode (with some students participating in-person and others joining remotely) whereas the other two schools operated fully remotely. In all sessions (30 in total), students were assigned individual work with algebra tutoring software. The teachers helped the students while the students were using the software. Experimenters attended each session remotely to provide

help where needed. Before the study, teachers worked through instructional materials about Tutti. During the study, teachers and experimenters had access to and monitored the dashboard.

5. **Testing hi-fi prototypes.** We conducted a series of prototyping sessions with early implementations of the dashboard to hone its design and usability. During these sessions, we interviewed teachers as they interacted with the tool. Four math teachers (grades 6 through 12) and one math director participated, with an average teaching experience of 19 years. One teacher was in Taiwan, the rest were in the US. All participants were asked to think aloud while performing 20 tasks using the dashboard's interactive capabilities. (The dashboard however was not updated in real-time during this study.) They were asked to report any potential problems they noticed, if they were to use it in class. We made many changes to the dashboard because of the findings.
6. **Conducting replay studies.** As a final way of honing the design of Tutti we conducted replay studies, that is, prototyping sessions during which teachers experienced some of the dynamic behaviors of the tool, though outside the real classroom environment. To create the dynamic behaviors, we replayed log data from a class of students (which captures their interactions with the tutoring software) through the dashboard in real time. The tool would update as it would if it were used during the real class session (cf. The Replay Enactments method [11]). In addition to testing usability, the study focused on how teachers would use the dashboard information to support their real-time decisions regarding whom to help. We also asked interview questions about desirable features in the tool. Three math teachers participated, all of whom had participated previously, with on average 19 years of teaching experience, teaching grades 7–12; one teacher taught special education classes. The data that was replayed came from a 6th grade class of 11 students, collected during the pilot study. As a result of the findings from this study, we made many changes to the dashboard.

5 Results from Teacher Interviews

We present insights from the early need finding activities (Research Activity #1).

Learning Process: "I Wish I Could see What They're Doing". All participating teachers described frustration when it came to identifying what students needed or how they were doing. Several of them noted the value in being able to see students' processes and actions as they normally would in their classroom. Several teachers working with MATHia (a commercially available tutoring system for mathematics) described how they used reports generated by the software and the software's live dashboard to understand if the students were working in the tutoring software, completed their assignments, and were on track to master content. They expressed a need for more detailed live information about what students were doing, as existing tools did not allow for remote monitoring. They described requesting or sharing screenshots with students over email or asking students to share their screens during individual meetings via teleconferencing software as a form of remote monitoring. However, not all students would respond or engage in one-on-ones with teachers.

Real-Time: "I Want to Know as Soon as Possible". Teachers wanted to get information about students as quickly as possible, so that they could correct problems immediately and provide timely praise. With remote instruction, teachers felt they could not identify and correct problems immediately as was possible in their normal classrooms. As a result, they could be missing moments of struggle until an assessment; some teachers strongly preferred reaching out and reteaching content to students before they experienced further frustration. Teachers also described missing the ability to quickly provide praise and support. One teacher remarked, "*Encouragement is a huge part of learning, saying, hey, you're moving in the right direction!*"

6 Insights from Storyboard Study

We present insights gained from the study with storyboards, Research Activity #2 listed above, which, as mentioned, focused on three aspects: Instructional context, displaying instances of student work, and technology options for teacher-student communication.

Viewing Specific Instances of a Student's Work is Valuable. Consistent with our earlier findings and those from the Lumilo project [11], the participating teachers unanimously valued the ability to follow students' processes in specific problem instances, both current and past. They viewed the different display options (Snapshots, Replay, Live Feed) as overlapping but complementary. The live problem view was deemed useful primarily for remote scenarios, as it may enable quick feedback and avoids the need for screen sharing by the student. Teachers felt that Replay (more so than Snapshots) enables them to investigate a student's challenges.

Time is of the Essence. Teachers (without prompting) evaluated whether the tool concepts depicted in the storyboards would help them operate efficiently. They found Snapshots attractive because they give quick insight, whereas they questioned whether they would have the time to use Replays or Live Feeds. Chat was viewed as the most efficient communication method, provided it would be well integrated with the dashboard and the tutoring software. The combination of Live Feed with drawing and chat was viewed as an efficient combination for use in remote scenarios. In live scenarios, teachers said they instead preferred to walk up to a student and talk.

Private Communications with Students are Highly Preferable. Consistent with past work on Lumilo [11], teachers valued tool and communication options that would safeguard students' privacy, in the sense that a student's struggles would not be known or visible to the entire class. For example, they did not want to show student names when displaying a Replay or when displaying the full dashboard to the class.

Teacher Attention Might Help Increase Participation in Class. Some teachers stated that students might be more motivated if they felt the teacher was keeping an eye on them - which the dashboard might help them do. They thought it might help to send "wake-up calls" (using chat or audio) to idling students or students misusing the system.

Teacher-Student One-on-One Communications via Chat Might be Useful Especially for Remote Students. Teachers thought audio communication with remote learners would be natural; they unanimously felt "normal interactions" would be possible in this manner (e.g., to redo a problem together with a student). One teacher mentioned that the use of chat in in-person scenarios might support multitask helping (help one student, write another; send a quick message and not interrupt students). They suggested having pre-defined, easily-customizable chat messages.

7 Observations from Remote Classroom Piloting

During the remote classroom pilots (Research Activity #4), teachers' activity with the dashboard's overview screen focused on checking which students were actively working on the tutor. (The Deep Dive screen had not been implemented yet.) During fully-remote sessions, many students had their webcam off, so teachers had no other easy way of ascertaining this information. The need to know who is working during educational technology use in fully-remote sessions has also been reported elsewhere [17]. The study revealed a need for a communication channel built into the dashboard when used for a remote or hybrid instructional mode, so teachers would not need a separate video conferencing tool (e.g., Zoom) to talk to a student.

We also observed that teachers did not make use of the notifications displayed on the dashboard's overview screen. We did not observe any instances, for example, where a teacher reached out to a particular student when a notification showed that the student was struggling. In this remote teaching context, teachers appeared to be occupied more with encouraging students to use the software and reaching out to students who did not make much progress (which could be gleaned from the progress table better than from the notifications) than reacting to indications of struggle. Indeed, during the study sessions, the teachers and experimenters exchanged many private messages regarding who is working on the software and who is not. This is not to say that notifications are not useful. Rather, their utility may depend on context, such as remote/in-person, and other factors (e.g., specific instructional goals teachers may have).

8 Key Design Features with Rationale

Following the storyboards, we narrowed our scope to focus on a smaller set of features that we expected to be useful for teachers. As (so far) teachers valued both Snapshot and Replay for in-class teaching, we decided to keep both, to further explore their complementary strengths through higher-fidelity prototyping. We put the Live Feed on hold, as the teachers said they would not use it often in person. Moreover, a live view had already been explored in past research on Lumilo [11]. We also dropped the communication options. Although some teachers saw use for them in live classrooms, we prioritized the display of analytics. Within this scope, the main design features are: (1) Two-screen design with easy navigation from class overview to student-level deep dive, (2) dual representations of students' status and recent behaviors (progress table and notifications) and (3) two ways of viewing instances of student work (Snapshots and

Replay). These features kept evolving during the subsequent activities (hi-fi prototyping and replay study). In the current section, we provide more detail about these features as they were at the end of the process. We also describe how they are grounded in data gathered during our interactions with teachers.

8.1 Two-Screen Dashboard Design

As mentioned, Tutti combines an *overview screen* (see Fig. 1) that provides information about each student in a class with a *deep dive screen* (Figs. 2, 3) that provides more detailed information about any given student's learning experiences. The information on the overview screen may help teachers get an initial sense of which students might need their attention (Fig. 1). To this end, the overview screen (a) summarizes progress through the problem sets with a simple visualization, (b) summarizes each student's learning state with indicators adopted from Lumilo, and (c) presents notifications of recent events regarding students' learning.

The information on the deep dive screen helps a teacher gain further insight into whether communication with the given student could be beneficial and what it might focus on (e.g., what skills, problem types, or errors a student may be struggling with). The deep dive screen may be a teacher's sole source of information about a student's work in remote scenarios. The deep dive screen provides information about the given student's progress through the assigned problem sets (Fig. 2, top right), with more detail available at the teacher's request including information about a student's skill mastery, areas of struggle (i.e., skills with substantial practice but low mastery; see Fig. 2., top left), and past problems (Fig. 2, bottom left). The problem list helps teachers gain insight into what problems were difficult for the given student, with information such as counts of errors, hints, and correct steps as well as the same indicators of progress and struggle that are used on the overview screen. Teachers can filter the problem list by skill, to select problem instances to inspect using either a Snapshot (Fig. 2, right) or a Replay of the solution (Fig. 3), as described in more detail below.

Teachers can access a student's deep dive screen in multiple ways, a design feature that make it easier to follow leads gathered from information on the overview screen. When the teacher clicks on a notification on the overview screen, the deep dive screen is initialized with information relevant to that notification, namely, the problem set and the problem the student was working on when the notification occurred. Similarly, when the teacher clicks on a student listed in the overview screen, the deep dive screen shows information related to that student's current problem set and problem.

The two-screen design (with a class overview screen and student-specific deep dive screen) is found in other teacher tools as well, including two dashboards used (like Tutti) in conjunction with AI-based tutoring software, Luna [25] and Lumilo [11]. These dashboards share the same raw data—tutor interaction data—and use analytics derived from that data. There are, however, some interesting differences regarding the information displayed on these dashboards, which could be attributed to the different use scenarios for which the dashboards were designed. For example, the overview screen of Luna, which is designed to support lesson planning, provides class aggregates, which are helpful when deciding what topic or examples to discuss in class. By contrast, Tutti only presents student-specific information on its overview screen, which is helpful when deciding which individual student to help. Further, where both Tutti and Lumilo present, on their student-specific deep dive screen, areas of struggle and examples of student work, Lumilo selects the examples for the teacher, whereas in Tutti the teacher has full control over which past problem instances to inspect.

8.2 State-Based and Event-Based Overview of Students' Learning

Although, on Tutti's overview screen (see Fig. 1), there is overlap between the information shown in the progress table and that captured in notifications, teachers preferred to have both representations. They use them for different purposes, and different teachers tend to rely to a greater or lesser extent on the notifications. In remote scenarios, the progress table shows which students are working with the tutoring software at any given moment, information they could not ascertain easily in other ways. The progress table also shows the current indicator values for each student.

The notifications draw teachers' attention to recent events. They are generated when there is a change of status in an indicator variable for a given student (e.g., a student enters the "idle" state or satisfies the definition for struggle). They also change (and are then displayed again at the top of the list) when a status has existed for a certain threshold amount of time. The notifications show how long the status has persisted (e.g., how long the student has been idle), the student, the problem set, and the specific problem the student is working on. Teachers can sort the notifications by student name or recency and can filter the notifications by student, type, or skill. Filtering and sorting can help teachers identify students who need help, as indicated by recent notifications, or simply go student-by-student to check on each student. Filtering by notification *type* makes it easy to view (say) all the struggle notifications and check whether they occur on similar math problems, or to identify all idle students and perhaps address all of them at once. Filtering the notifications by skill helps to ascertain whether there are class-level problems related to any specific skill. (Perhaps a brief mini lecture to the whole class is in order.) Some teachers mentioned that the notifications could help them get students back to work quickly (e.g., when there is no strong evidence of struggle, only an idle indicator or system misuse indicator). One teacher indicated they wanted the notifications to be always visible (i.e., on both the deep dive and overview screens).

8.3 Snapshots and Replay to View Examples of Student Work

Many teachers indicated that viewing specific examples of a student's work is a key way for them to discover what that student finds difficult. Initially, we thought of Snapshots and Replays of student work as *alternative* designs for meeting this need; we expected that teachers would gravitate towards one or the other. We found, however, that *both* were attractive options to teachers, with Replay being slightly preferred. Although an annotated Snapshot would appear to be faster (and time is of the essence, as discussed above), an argument in favor of Replay, in the words of one teacher, is that it is more like what you would see if you interacted with a student.

Snapshots take up more screen real estate, compared to Replay, as the problem steps are annotated with "action icons" that represent each hint, error, and correct action (see Fig. 2, right). The main challenges in designing the Snapshot screen were placing these annotations so they do not cover up the problem steps and showing the order in which the student's actions happened. We tried multiple concepts (e.g., numbering actions and representing multiple similar actions with a single icon). However, showing a row of single-action icons row seemed the most straightforward and easy to interpret. This solution, however, communicates the order of student actions only for tutor interfaces in which the order of problem steps is fixed (e.g., the equation-solving tutor shown in Figs. 2 and 3), and not for tutor interfaces in which the order of steps can vary. At the suggestion of teachers, tapping or hovering on an action icon is used as an intuitive way to show specific errors or hints. We used the same color coding for hints, errors, and correct entries in all parts of the deep dive screen. Snapshots (unlike Replay, for technical reasons) can be applied to the student's current problem, although without automatic live updating. A "Full screen" option for Snapshots helps allocate more screen space, which is useful for large tutor interfaces. The full-screen mode hides the student name, which is useful when the teacher wants to project a problem solution for the whole class to see and discuss. We also added a hide name button (top left, Fig. 2). Teachers strongly wanted to maintain student anonymity when sharing student work.

Key design decisions in creating the Replay functionality were, first, to model the controls after those often used in media players (e.g., video/music); second, to make the bar draggable and minimizable so it does not obstruct the teacher's view; and third, to use a fixed duration for each replayed action (2s; teachers preferred this speed). Teachers commented that they would use Replay for reviewing problem solutions with students, individually or with the whole class. One teacher commented that they would use Snapshots with the more advanced students, as doing so would be efficient, whereas for less advanced students, Replay would be more concrete and recognizable. During the replay study (Research Activity #6) Replay was a popular feature, although teachers also looked at Snapshots often and expressed a liking for them. Teachers suggested several new use cases for Replay. One teacher thought Replay might help them get to know new students more quickly. Another thought it might work as a homework tool for students who are behind. Finally, one teacher suggested Replay together with recording a proof of help might be used in parent-teacher conferences.

9 Discussion and Conclusion

Analytics-based support tools for teachers who run personalized classrooms with AI-based software pose unique design challenges, yet there is relatively little general knowledge regarding the design of such tools. To address this challenge, we created Tutti through user-centered design and prototyping. Teachers found three main design features to be helpful: A design with both a class overview screen and a student-level deep dive screen, multiple views of data about a class of students (a progress table and notifications), and two ways of looking at specific instances of problems solved by a given student, either in the form of a Snapshot (with annotations that show hints and errors) and a Replay of all student step attempts and hints.

The work adopts several design elements from Lumilo, a mixed-reality tool that helps teachers help their students during self-paced, personalized learning [11], but is also different. Tutti uses commonly available hardware (e.g., tablet computers). It adds interactive exploration of students' learning experiences; it features multiple views of student information, ways of quickly finding past problem instances where a student struggled, and teacher control over which past problem instances to inspect. The work confirms the value of the shared design elements and suggests there is value in the new elements (e.g., Snapshots and Replay).

The work's key contributions include new knowledge regarding the design of analytics tools for real-time helping of students during self-paced learning with AI-based tutors, grounded in data about teachers' needs. The work also contributes new insights into teachers' needs during self-paced technology-enhanced learning and how these needs vary by context, such as whether instruction is in-person or remote. Further, the work provides new insight into how the importance of design features of real time analytics-based teacher tools varies by context. Regarding the generality of the work, Tutti is designed for use with any tutor built within CTAT + Tutorshop [3]. We tried out Tutti with several tutors built within this infrastructure and found it can be useful with them, although further generality testing is in order. More broadly, Tutti might be used with any tutoring system that supports stepwise problem-solving practice and tracks student learning of detailed skills. Perhaps some design features could be useful with other forms of technology-enhanced problem-solving practice as well.

Although the design of Tutti is grounded in extensive data of teachers' needs and an early version was pilot-tested in remote teaching scenarios, more classroom piloting with the complete tool is needed. A second limitation is that the design of Tutti is not grounded in data about *students'* needs and preferences. It may help to study these needs for example through Holstein et al.'s multi-stakeholder iterative speed dating method [12]. Finally, it will be interesting to test, in a classroom study how students' learning experiences and outcomes are affected when the teacher uses the dashboard (cf. [13]).

Knowledge regarding the design of support tools for teachers, such as that generated in the current project, may have both practical and theoretical value: Practically, it may serve as a foundation for future projects. Theoretically, it enhances our understanding of how to harness the power of analytics for use by teachers in specific use scenarios.

Acknowledgments. The research was supported by US Dept. of Education (IES) grant R305A180301 and NSF (IIS) grant 1822861 We gratefully acknowledge their contributions. Opinions expressed in the paper are those of the authors, not the sponsor.

References

1. Ahn, J., Campos, F., Hays, M., Digiacomo, D.: Designing in context: reaching beyond usability in learning analytics dashboard design. J. Learn. Anal. **6**, 70–85 (2019)
2. Ahuja, K., Kim, D., Xhakaj, F., Varga, V., Xie, A., et al.: EduSense: practical classroom sensing at scale. Proc. ACM Interact Mob. Wearable Ubiq. Technol. **3**, 1–26 (2019)
3. Aleven, V., McLaren, B., Sewall, J., van Velsen, M., Popescu, O., et al.: Example-tracing tutors: intelligent tutor development for non-programmers. Int. J. Artif. Educ. **26**, 224–269 (2016). https://doi.org/10.1007/s40593-015-0088-2
4. An, P., Holstein K., d'Anjou B., Eggen B., Bakker S.: The TA framework: designing real-time teaching augmentation for K-12 classrooms. In: Proceeding 2020 CHI Conference, pp. 1–17. New York, ACM (2020)

5. Arroyo, I., Woolf, B., Burleson, W., Muldner, K., Rai, D., et al.: A multimedia adaptive tutoring system for mathematics that addresses cognition, metacognition and affect. Int. J. Artif. Educ. **24**, 387–426 (2014). https://doi.org/10.1007/s40593-014-0023-y
6. Baker, R.S., Yacef, K.: The state of educational data mining in 2009: a review and future visions. J. Educ. Data Mining **1**, 3–17 (2009)
7. du Boulay, B.: Recent meta-reviews and meta–analyses of AIED systems. Int. J. Artif. Educ. **26**, 536–537 (2016). https://doi.org/10.1007/s40593-015-0060-1
8. Corbett, A.T., Anderson, J.R.: Knowledge tracing: modeling the acquisition of procedural knowledge. User Model User-adapt. Interact. **4**, 253–278 (1995). https://doi.org/10.1007/BF01099821
9. Gupta, A., et al.: Affective teacher tools: affective class report card and dashboard. In: Roll, I., McNamara, D., Sosnovsky, S., Luckin, R., Dimitrova, V. (eds.) AIED 2021. LNCS (LNAI), vol. 12748, pp. 178–189. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-78292-4_15
10. Hanington, B., Martin, B.: Universal methods of design expanded and revised. Rockport Publishers (2019)
11. Holstein, K., McLaren, B.M., Aleven, V.: Co-designing a real-time classroom orchestration tool to support teacher–AI complementarity. J. Learn. Anal. **6**, 27–52 (2019)
12. Holstein, K., McLaren, B.M., Aleven, V.: Designing for complementarity: teacher and student needs for orchestration support in AI-enhanced classrooms. In: Isotani, S., Millán, E., Ogan, A., Hastings, P., McLaren, B., Luckin, R. (eds.) AIED 2019. LNCS (LNAI), vol. 11625, pp. 157–171. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-23204-7_14
13. Holstein, K., McLaren, B.M., Aleven, V.: Student learning benefits of a mixed-reality teacher awareness tool in AI-enhanced classrooms. In: Rosé, C.P., et al. (eds.) AIED 2018. LNCS (LNAI), vol. 10947, pp. 154–168. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93843-1_12
14. Kelly, K., Heffernan, N., Heffernan, C., Goldman, S., Pellegrino, J., Goldstein, D.S.: Estimating the effect of web-based homework. In: Chad Lane, H., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS (LNAI), vol. 7926, pp. 824–827. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39112-5_122
15. Koedinger, K.R., Aleven, V.: Multimedia learning with cognitive tutors. In: Mayer, R., Fiorella, L. (eds.) Cambridge Handbook of Multimedia Learning, 3rd edn., pp. 439–451. Cambridge University Press, Cambridge, UK (2022)
16. Knoop-van Campen C.A.N., Wise A., Molenaar I.: The equalizing effect of teacher dashboards on feedback in K-12 classrooms. Interact. Learn. Env. 1–17 (2021)
17. Nagashima, T., Yadav, G., Aleven, V.: A framework to guide educational technology studies in the evolving classroom research environment. In: De Laet, T., Klemke, R., Alario-Hoyos, C., Hilliger, I., Ortega-Arranz, A. (eds.) EC-TEL 2021. LNCS, vol. 12884, pp. 207–220. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-86436-1_16
18. Olsen, J.K., Rummel, N., Aleven, V.: Designing for the co-orchestration of social transitions between individual, small-group and whole-class learning in the classroom. Int. J. Artif. Educ. **31**, 24–56 (2021). https://doi.org/10.1007/s40593-020-00228-w
19. Pane, J.F., Griffin, B.A., McCaffrey, D.F., Karam, R.: Effectiveness of cognitive tutor algebra I at scale. Educ. Eval. Policy Anal. **36**, 127–144 (2014)
20. Ritter, S., Anderson, J.R., Koedinger, K.R., Corbett, A.: Cognitive tutor: applied research in mathematics education. Psychon. Bull. Rev. **14**, 249–255 (2007). https://doi.org/10.3758/BF03194060
21. Ritter S., Yudelson M., Fancsali S.E., Berman S.R.: How mastery learning works at scale. In: Proceeding 2016 ACM Conference on Learning @ Scale, pp. 71–9, New York ACM (2016)
22. VanLehn, K.: The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. Educ. Psychol. **46**, 197–221 (2011)

23. Wang, S., Christensen, C., Cui, W., Tong, R., Yarnall, L., et al.: When adaptive learning is effective learning: comparison of an adaptive learning system to teacher-led instruction. Interact. Learn. Env. 1-11 (2020). https://doi.org/10.1080/10494820.2020.1808794
24. Wood, H., Wood, D.: Help seeking, learning and contingent tutoring. Comput. Educ. **33**, 153–169 (1999)
25. Xhakaj, F., Aleven, V., McLaren, B.M.: Effects of a teacher dashboard for an intelligent tutoring system on teacher knowledge, lesson planning, lessons and student learning. In: Lavoué, É., Drachsler, H., Verbert, K., Broisin, J., Pérez-Sanagustín, M. (eds.) EC-TEL 2017. LNCS, vol. 10474, pp. 315–329. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66610-5_23
26. Zimmerman, J., Forlizzi, J.: Speed dating: providing a menu of possible futures. She Ji J. Des. Econ. Innov. **3**, 30–50 (2017)