https://doi.org/10.2352/EI.2022.34.1.VDA-413

CoursePathVis: Course Path Visualization Using Flexible Grouping and Funnel-Augmented Sankey Diagram

Brendan J. O'Handley, Morgan K. Ludwig, Samantha R. Allison
Michael T. Niemier, Shreya Kumar, Ramzi Bualuan, Chaoli Wang
University of Notre Dame

Abstract

We present CoursePathVis, a visual analytics tool for exploring and analyzing students' progress through a college curriculum using a Sankey diagram. Focusing on four student cohorts in a department, we group students in multiple ways (by their AP courses, term courses, and a user-specified funnel course) to comprehensively understand the data. CoursePathVis helps us identify patterns or outliers that affect student success with these flexible grouping techniques and the funnel-augmented Sankey diagram. Three stakeholders from the same department formulate design requirements and provide an ad-hoc evaluation.

Introduction

Within a university, a college or department desires all students to succeed in their coursework and complete their degree in time. However, there can be many factors why this does not happen. Common factors include student preparedness (e.g., their advanced placement or AP[1] credentials), course difficulty (e.g., particularly challenging required courses), and personal reasons (e.g., health and mental issues). Suppose we can group hundreds of students in a major and visualize how they progress through their courses. Then, we could examine potential issues and take action to help students succeed.

This motivates us to design and develop CoursePathVis, a visual analytics tool for analyzing and visualizing how students take courses over terms toward their degree completion. The data for this study is an anonymized collection of student records from a computer science and engineering (CSE) department, including four cohorts (i.e., the classes of 2019, 2020, 2021, and 2022) with an average of 143 students in each cohort. Each student record contains various attributes, including courses the student has taken, the term in which the course was taken, the grade for each course, the student's term GPA, cumulative GPA, college, and whether the student had taken pre-college (outside the department) courses. We consider a total of 27 required courses for the CSE major.

To analyze the collection of student records, we design different ways to group students based on their AP courses and term courses. These functions allow evaluating students' academic success as they progress course by course or term by term. On top of AP courses or term courses based on student grouping, we augment the resulting visualization via a new feature called a *funnel* course. A funnel course is usually chosen from gateway (i.e., the first credit-bearing college-level course in a program of study) or required courses, which allows us to combine initial student groups into a new single group, impacting the downstream flows of the visualization. These initial groups can be displayed using a histogram popup over the funnel course (e.g., `Fundamentals of Computing` shown in Figure 6 (a)). As such, we can generate comprehensive visualizations for visual reasoning, helping stakeholders identify patterns and discover insights.

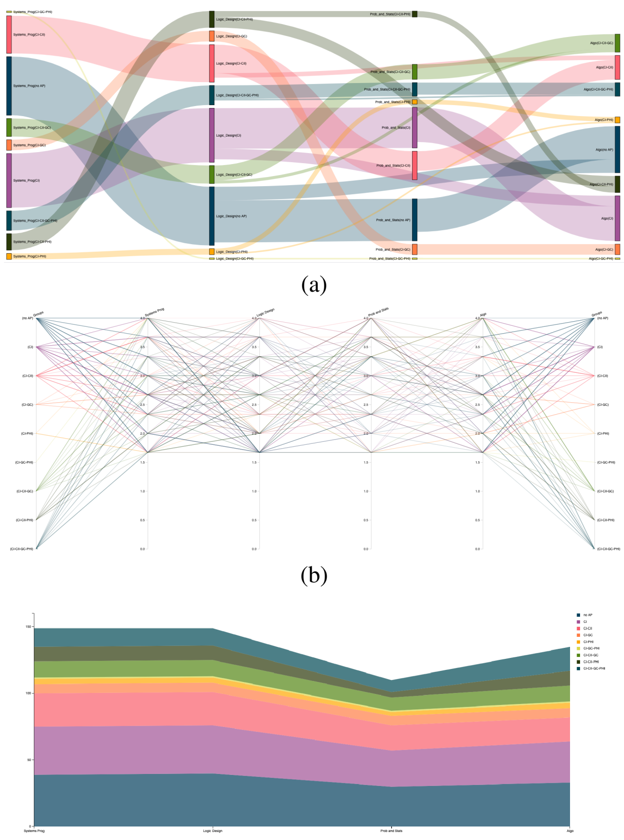
[1] AP refers to "a program in the United States and Canada created by the College Board which offers college-level curricula and examinations to high school students." [1]

Related Work

Learning analytics dashboards. Stakeholders such as instructors, academic advisors, and administrators have previously used individual student and course data to identify areas of struggle or success. For example, learning analytics dashboards [16, 24] display various information such as login frequency, time on task, clickstream, and tool/resource usage by students in an online learning environment. Early research efforts focused on highlighting potential students at risk of academic failure [4, 2, 3]. Later, researchers developed student-facing dashboards (i.e., dashboards used by students) to increase students' self-awareness, promote positive behavior change, and ultimately enhance their academic achievements [22, 5, 18]. Although the majority of existing learning analytics dashboards are for instructors and students, a few can assist administrators in strategic decision-making [19, 17, 6]. Some learning analytics visualizations benefit different user groups, including students, instructors, and administrators [20, 11, 8]. Besides uncovering individual students' learning behavior in an isolated context, this type of visualization can also help identify how students form groups and interact with each other in a social network context [27, 9, 7].

Course program visualization. Focusing on a single course is not enough when considering the "flow" of all students through their entire academic study. Moving from the *course-level* to *curriculum-level* analysis entails a different set of design requirements. While there are recent works that focus on visual analytics of student progression in the higher education setting [14, 15], the proposed CoursePathVis aims to understand how students progress through the curriculum and identify common patterns and outliers for stakeholders to make better-informed decisions or recommendations. Our work is closely related to course program data visualization [23, 21]. Instead of using fine-grained trajectories followed by clustering and composition [23] or a coarse-grained Sankey-like radial graph showing student progression in a university-wide setting [21], we employ the funnel-augmented Sankey diagram to investigate course paths taken by students within a department.

Event sequence data visualization. The large-scale, high-dimensional, and heterogeneous nature of event sequence data has motivated many recent research activities in the visual analysis of event sequence data [13]. Examples include visualizations based on the chart, timeline, hierarchy, Sankey, and matrix. Prior work on Sankey-based event sequence data visualization includes Outflow [26], DecisionFlow [10], CAVA [28], and EventThread [12]. Our design of the "funnel" course is similar to the notions of "sentinel" [25] (e.g., the first occurrence of a given symptom type) or "milestone" [10] (e.g., temporal query constraints) events to aid with aligning temporal event sequences. However, instead of sequence alignment, we leverage the funnel course to investigate its impact on students' course progress and performance by channeling students into the designated course and presenting filtered visualization results.



(a) (b) (c)

Figure 1. *Comparison of three design alternatives. (a) to (c): Sankey diagram, parallel coordinates, and stacked area chart. Students are grouped by AP courses taken. The horizontal axis or column represents individual courses (inner four axes for parallel coordinates). Vertically, we display student groups, course grades, and student counts, respectively, for these three visual mappings.*

Design Requirements

CoursePathVis was driven by a CSE department's need to look at how undergraduate students take required courses throughout the curriculum. The department has offered B.Sc. in Computer Science and B.Sc. in Computer Engineering degrees and has started to offer B.A. in Computer Science degrees. With the increase in degree offering and steady enrollment growth, it is crucial to investigate past students' course paths to identify strengths and weaknesses to serve current and future students better.

As such, we had several discussions with three stakeholders from the CSE department: the Director of Undergraduate Studies **S1**, the Chair of Undergraduate Curriculum Committee **S2**, and an introductory course instructor **S3**. They would like the proposed CoursePathVis to fulfill the following needs: (1) providing an overview of how student cohorts take the required courses throughout the terms (e.g., the common paths and the patterns over terms); (2) enabling the investigation of how students with different preparedness levels take the introductory courses and the implications to their academic performance and time to degree; (3) studying how students perform for a required course and how this would impact their subsequent course taking. Based on these needs, together, we formulate the following design requirements: **R1:** *Provide an overview of course paths.* The tool should provide a good overview for users to understand the overall patterns and complexity to start with, triggering the consequent exploration by grouping and filtering. **R2:** *Group students flexibly.* The tool should enable users to group students by AP courses (math, science, or all AP courses) or term courses to compare different student groups visually. To help investigate student performances, we should show the average GPA of student groups and allow ordering groups based on the GPA. **R3:** *Filter courses or students.* The tool should allow users to filter the data by student graduation year, grade level (freshman, sophomore, junior, senior), or courses. Furthermore, users need to filter the visual results based on the range of group size to spot common patterns (large groups) and outliers (small groups). **R4:** *Enable the funnel course.* The tool should allow users to funnel multiple student groups into a single course to study how students get through a gateway course [23]. Displaying these student groups' grade distributions further helps investigate their performances and possible impact on the subsequent course taking.

Design and Implementation

Design choices. We consider three popular visual mappings: Sankey diagram, parallel coordinates, and stacked area chart, to display sequence data. As shown in Figure 1, parallel coordinates do not show student groups in an aggregated manner, making it challenging to estimate student counts and track student groups across the axes. The stacked area chart is good at revealing group-wise information based on AP courses (horizontal: individual courses, vertical: student counts). However, it has difficulty showing term courses which are varying over terms (horizontal: individual terms, vertical: term courses) and does not support sorting by GPA. The Sankey diagram does not have such limitations (refer to Figure 2) and is the most suitable choice for our application.

Grouping students by AP courses. One way of student grouping is based on AP courses. This is often needed as the stakeholders would like to examine the impact of AP courses on student course-taking and their resulting performance differences. We first identified students who had certain AP credits by the courses they had taken. We grouped students according to the various AP course combinations. The certain AP credits

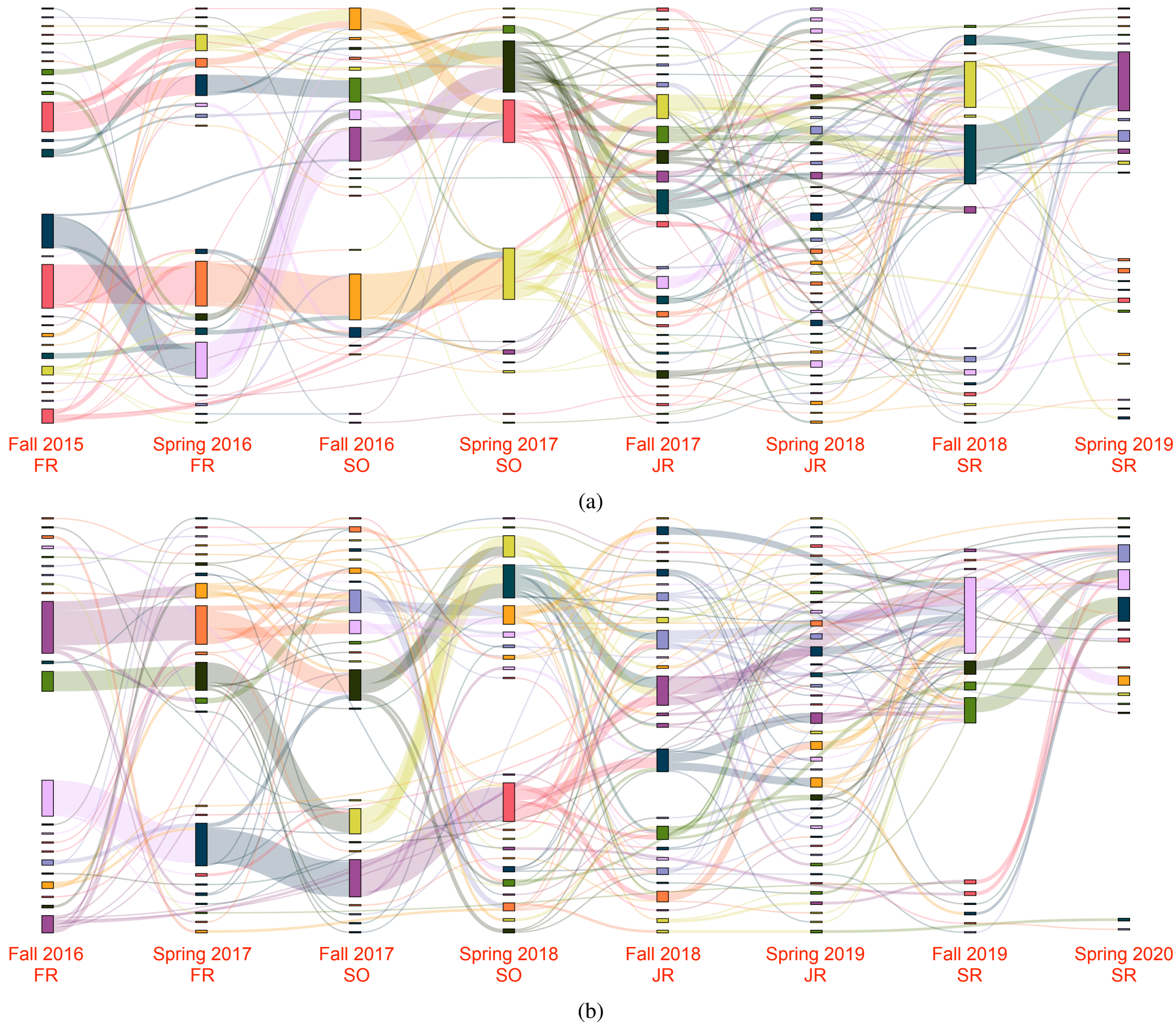


Figure 2. *Comparison of two graduation years (a) 2019 and (b) 2020 using "Courses by Term," which shows how student groups in the CSE major bundled courses in each term. Sorting the student groups by GPA is turned on. Node labels are hidden for clarity.*

include `Calculus I`, `Calculus II`, `Physics I`, `Physics II`, and `General Chemistry`. One example of a course combination would be a group of students who have AP credits in `Calculus I` and `General Chemistry`. We then identified the list of courses in the sequence they were taken. We ordered these courses and drew them as nodes (from left to right) in the Sankey diagram. Each student group uses a different color for differentiation. Finally, we created the links between nodes by iterating through the previous collections of groups and uniquely identifying the student group's flow from course to course, term by term. Note that this is the exact flow for each student within a group from course to course. Each student within a group took the course. They are then further grouped in a link that connects to the next course. Besides grouping students based on all AP courses, we split AP courses into math courses and science courses. Each of these additional student groupings reduces groups' variations as there are fewer courses to consider. We further provide filtering AP groups, allowing users to filter overprepared or underprepared student groups (i.e., incoming students with more or less AP credits than anticipated). For example, when "Math AP Courses" is selected for grouping students, users can further choose (no math AP), `Calculus I` (CI), or `Calculus I` and `Calculus II` (CI-CII) to filter AP groups.

Grouping students by term courses. Another way of student grouping is based on term courses. This allows the stakeholders to investigate how students taking a course group would impact their subsequent course group taking and figure out the most or least popular course group to take each term (a helpful feature for course recommendation). We identify students who took the same group of courses within the same term. For example, many freshmen took `Calculus I`, `Engineering`, and `General Chemistry` in their first term. In the Sankey diagram, these students are grouped to form a node along the first column. Each student may then take another set of courses for the

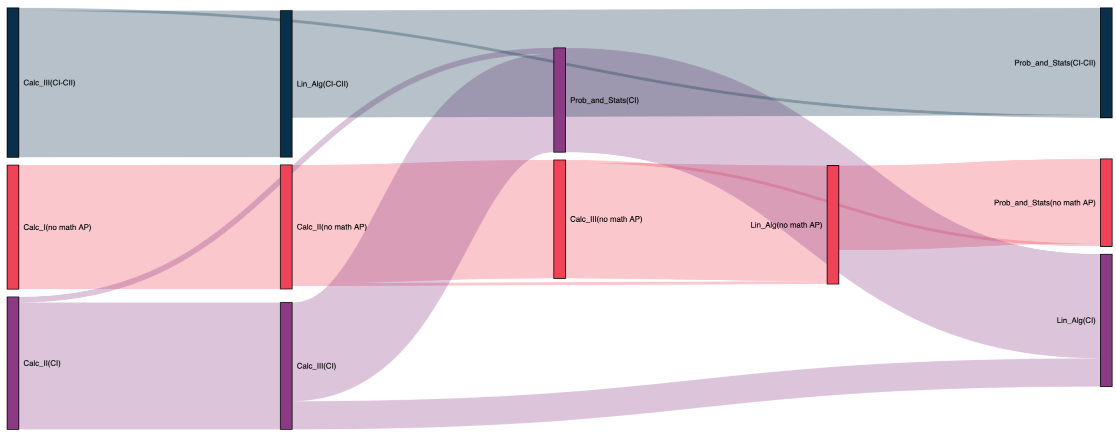


Figure 3. *Grouping students based on "Math AP Courses." Only math courses are displayed in the Sankey diagram. Sorting the student groups by GPA is turned on.*

next term, so the group splits and flows into the next set of term courses. We achieve such a grouping by first iterating through the student records and creating all unique term-course groups. We then associate students with these groups to access their data, such as cumulative GPA and more. Next, the term-course groups are linked together, ordered by term. This processing allows users to see all scenarios of courses taken, term by term, by all students within a major.

Grouping students by a funnel course. Besides grouping students based on AP courses or term courses, we allow additional grouping by a funnel course. There are certain required courses that all students should take in a given term. Users can funnel existing student groups (either by AP or term courses) into such a gateway course. This enables the stakeholders to explore how a particular funnel course impacts subsequent course taking or how students progress up to the funnel course. When the funnel course is selected, it builds a larger collection of all existing groups that include the funnel course and replaces each with the new funnel course group. Each AP course or term-course group is preserved within the funnel course group, along with a histogram showing each unique group's GPA for the funnel course. The funnel course comes with a cutoff feature. When the cutoff is disabled, we continue the Sankey diagram flow out from the funnel course as usual. When the cutoff is enabled, we hide groups after the funnel course. The cutoff does not function completely in some term-course group displays because students may have taken the funnel course in various terms due to transferring in or changing majors. In this case, we only generate the funnel course for the chosen course's first occurrence.

Results and Discussion

We present selected CoursePathVis results to show that it has met the design requirements and offered insight into the curriculum. In addition, we have deployed a version of CoursePathVis at `https://www.nd.edu/~cwang11/cpvis/`, where users can load randomized data to test the functions.

We meet **R1** by providing an overview of all course paths for a curriculum. Figure 2 shows a comparison of all course paths for two cohorts of students, graduating in the years 2019 and 2020, respectively. Using the "Courses by Term" grouping, each column in the Sankey diagram represents a term, and each node in a column represents a term-course group. The width of the link or flow between two nodes in the adjacent columns indicates the number of students taking both term-course groups. We can see that there are larger ("taller") nodes in the first two years of study for each cohort, showing that students group similar required courses together for these terms (i.e., they have less flexibility in taking elective courses). As students move to their junior and senior years, we see the average size of nodes decrease. This indicates that students have a more varied way of grouping their required courses by term (i.e., they have more flexibility in taking elective courses as more required courses are completed). One notable difference between 2019 and 2020 is in the sophomore spring term. In 2019, there are three large nodes, while 2020 has four nodes of slightly smaller sizes for the same term. This implies more variety of required courses offered in the sophomore spring term for 2020, which is true as the department started to provide more required courses in both terms from the 2017-2018 academic year.

CoursePathVis meets **R2** by grouping students flexibly. Figures 2 and 3 show flexible grouping by term courses and AP courses, respectively. As shown in Figure 3, using the "Math AP Courses" grouping, we form three student groups: no math AP credit (pink), `Calculus I` credit (purple), and `Calculus I` and `Calculus II` credit (teal). In the Sankey diagram, nodes from left to right in the same group (shown in the same color) represent the courses taken by the student group in sequence. We order the nodes along each column by their average GPA from top to bottom (mousing over each node displays the GPA information). We can see that students with `Calculus I` and `Calculus II` credit (teal) maintained the highest GPA overall. We also notice that among these courses, students with no math AP credit (pink) took more courses overall.

We support filtering courses or students to meet **R3**. As shown in Figure 4, we first select students from the graduation year of 2019. We then add another filter to select only courses students took in their freshman and sophomore years. After that, we turn on the grouping of students by "All Math courses." We can see that underprepared students (the teal group shown at the top) take significantly more math courses than overprepared students

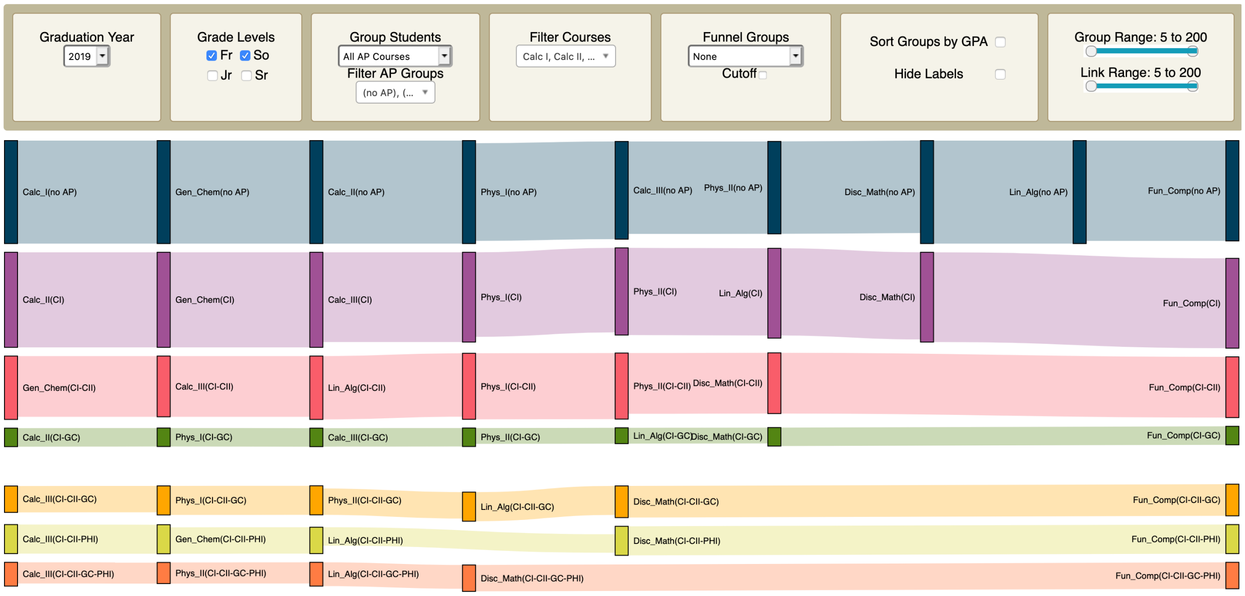


Figure 4. *A variety of filters with grouping students by "All AP Courses." Grade levels are filtered by selecting only "Fr" (freshman) and "So" (sophomore) years. Nine courses from 27 required courses are selected. Sorting the student groups by GPA is turned off. The display range for both group and link is narrowed to [5, 200] students.*

(the orange group displayed at the bottom) in their first two years. By hovering over the nodes, we can also see the average GPA for each group to compare similar courses by different groups. Academic advisors can then use this knowledge to advise students concerning which courses to take and when to take them.

As another example highlighting **R3**, Figure 5 shows the comparison of the most and least popular course paths for students graduating in 2022. From Figure 5 (a), we can see that the most popular term-course groups for the four terms from the freshman and sophomore years are, respectively, (`Calculus I`, `Engineering`, `General Chemistry`), (`Calculus II`, `Physics I`), (`Calculus III`, `Discrete Math`, `Fundamentals of Computing`, `Physics II`), and (`Data Structures`, `Linear Algebra`, `Logic Design`, `Systems Programming`). Clicking on the links shows that there are 36, 36, and 37 students taking the course paths across these four term-course groups, which accounts for around 25% of the cohort. In contrast, there are many more least popular course paths, as shown in Figure 5 (b). Students taking these outlier course paths could be either ahead of or behind schedule. Some students are "ahead of schedule," taking a required course (e.g., `Calculus III`, `Discrete Math`, or `Fundamentals of Computing`) in their first term, which the majority takes in their third term. Other students are "behind schedule," either taking a smaller set of required courses in their first term or catching up on a course such as `Writing and Rhetoric` which the majority has taken as a pre-college course. Furthermore, we also see multiple cases where students take a non-least popular term-course group (the corresponding node is hidden) in one term and then take a least popular term-course group (the diverging links are shown) next term. This pattern can be observed throughout the terms, indicating fairly dynamic changes of term-course groups. Given these findings, academic advisors could further track these students to understand these least popular course paths and provide a more informed recommendation for current and future students.

We also meet **R4** by enabling the funnel course. Figure 6 shows the term-course flexible grouping funneled into `Fundamentals of Computing` for the cohort graduating in 2019. In addition to showing how term-course groups funnel into this course, mousing over the funnel course displays the histogram that shows bins of grades (`A`-`F`) for each term-course group for the funnel course. We see that the largest group of students (42 students) took the courses `Calculus III`, `Discrete Math`, and `Physics II` simultaneously as the funnel course. Within that group, more students received a `C` or lower for the funnel course. We also see that 19 out of 20 students who paired only `Discrete Math` with the funnel course received an `A` in `Fundamentals of Computing`. This finding may help academic advisors better guide students in planning their course paths to achieve a more desirable GPA. In Figure 7, we show another example of funneling the course `Data Structures` for the same cohort. Grouping students by all AP courses is employed. Unlike Figure 6 where the cutoff does not function completely, in this case, the cutoff hides groups after the funnel course. Academic advisors can observe the primary course paths leading to `Data Structures` and investigate the GPA variations along these paths.

Limitations. Currently, CoursePathVis has two limitations. First, simultaneous comparison of different years is not available due to limited screen space. Users need to switch between different settings and get screenshots for side-by-side comparison. We can resolve this by developing a juxtaposed view to enabling direct comparison. Second, the current tool may not scale well if we include elective courses and include all student data at the college level. More research is needed to reduce visual occlusion and clutter while supporting large data input and adaptive visual reasoning.

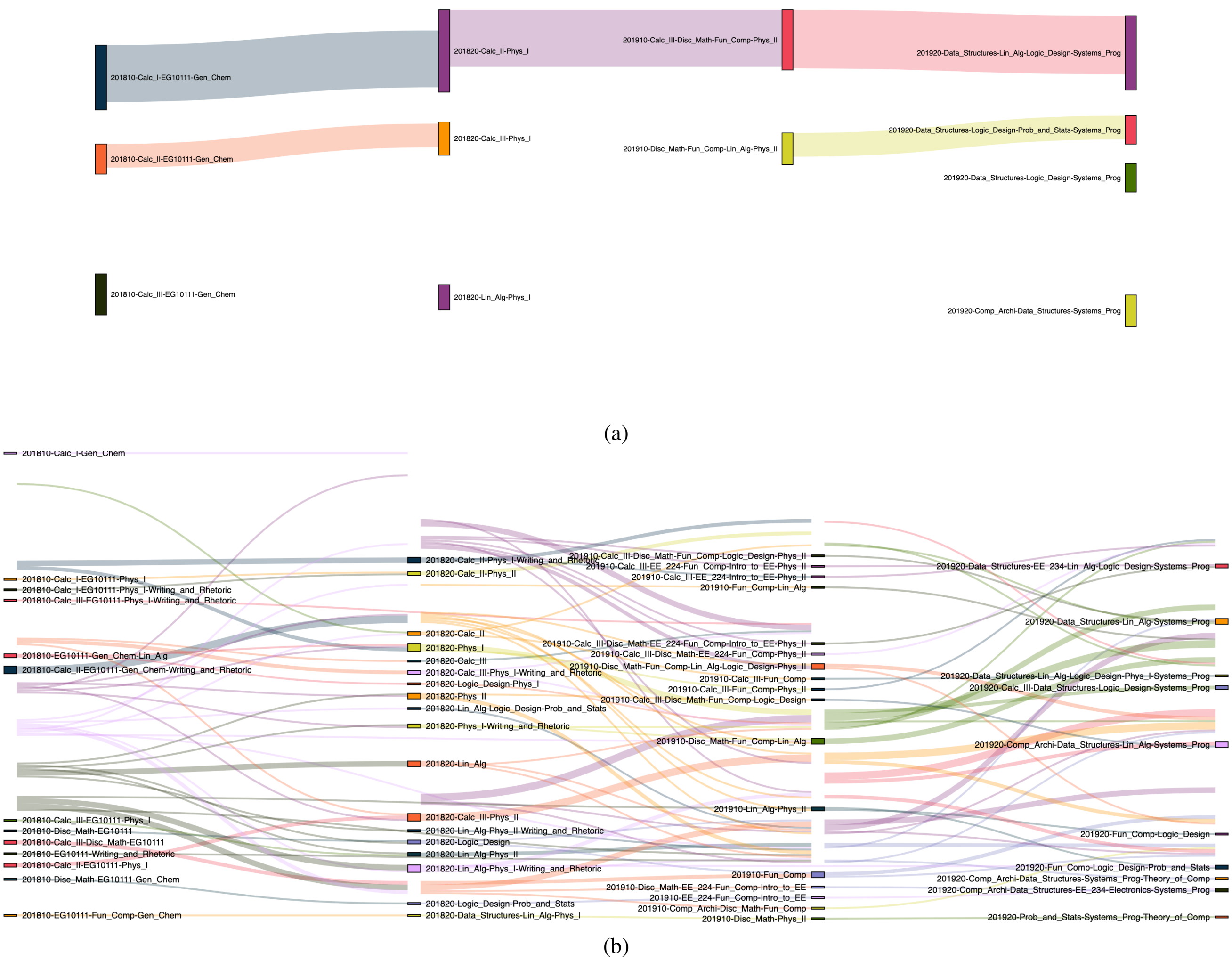
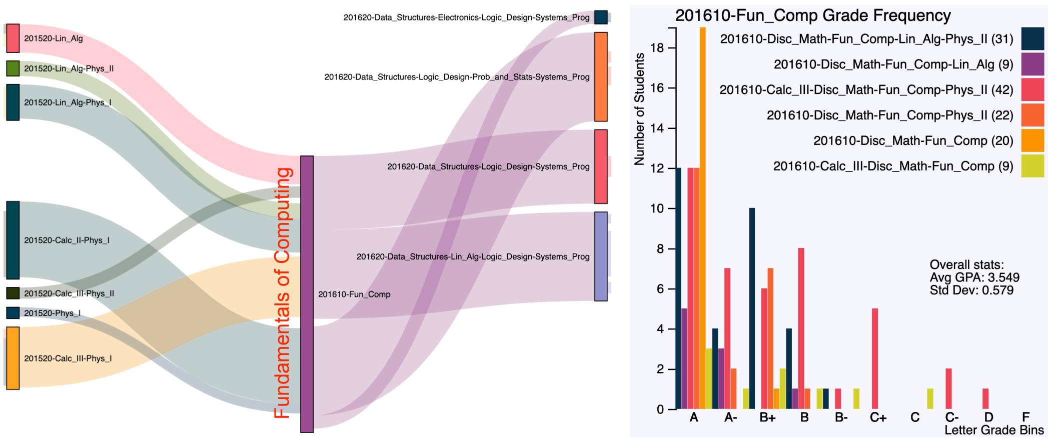


Figure 5. *(a) and (b): the most and least popular course paths for students graduating in 2022. Grade levels are filtered by selecting only "Fr" (freshman) and "So" (sophomore) years. Sorting the student groups by GPA is turned off. The display ranges for both group and link are [15, 200] and [1, 4] students, for the most and least popular course paths, respectively.*

Ad-Hoc Evaluation

CoursePathVis was evaluated by the three stakeholders (**S1**: the Director of Undergraduate Studies, **S2**: the Chair of Undergraduate Curriculum Committee, and **S3**: an introductory course instructor) who previously formulated design requirements. We first demonstrated the tool and showed the results gathered in the paper to them. Then, they used the tool for free exploration (each lasted about 30 minutes). We found they could navigate the Sankey diagram and interpret the visual results even though they were not familiar with this visualization form. In the end, we asked them to answer three questions: (1) does CoursePathVis meet the requirements? (2) what insights were you able to find using CoursePathVis? and (3) any suggestions to expand or improve CoursePathVis? Their review is as follows.

S1 stated that "*As the director of undergraduate studies in the department, I often have to deal with evaluating and advising students based on the path of courses they have taken, both in high school and then in college. The CoursePathVis tool provides many of the features that make it much easier to advise the students better.*" **S3** added that "*Some of the features we were looking for had to do with giving us abilities to (1) assess the flexibility of pathways in our curriculum, (2) better advise students with different levels of high school preparedness, (3) spot potential problem areas in the curriculum and assess them for their workload, and (4) help in finding if there are ideal combinations of courses to improve student success. The tool helps serve more significant questions, e.g., do general engineering course grades affect or predict grades in independent computing courses?*" **S2** pointed out that "*With a quick assessment, I found that AP experience, especially for Calculus courses, was a predictor for higher grade performance in freshman engineering courses.*" This confirmed the insight previously discovered by an assistant dean of the College of Engineering (COE) who had not used CoursePathVis. **S3** stated that "*Surprisingly, I found that AP experience also correlated with lower grade performance in downstream computing courses that did not depend on any AP courses' knowledge.*" This led her to



(a) (b)

Figure 6. *(a): the cropped view of the Sankey diagram showing the funnel course* `Fundamentals of Computing` *and various filters with grouping students by term courses. All grade levels and courses are selected. The cutoff is enabled. Groups are sorted by GPA, and the range for both group and link is set to [5, 200]. (b): the histogram of term-course groups GPA for the funnel course.*

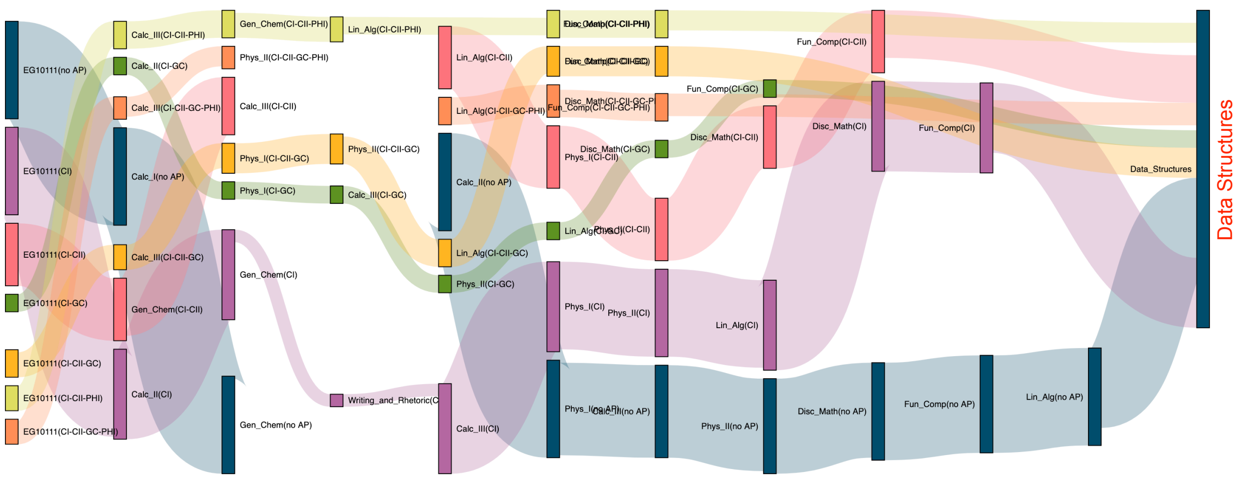


Figure 7. *The funnel course* `Data Structures` *with grouping students by all AP courses. All grade levels and courses are selected. The cutoff is enabled. Groups are sorted by GPA, and the range for both group and link is set to [5, 200].*

investigate whether term-course workload affects grades. Using the tool, she examined what course combination students with AP Calculus experience typically took in their first two years and compared that with non-AP students, using different courses as funnels or just staying with the term-course view. **S3** added that "*I found that students having AP experience could skip some courses that led to busy terms. They took a less heavy combination of intensive math- and computing-courses in their sophomore year, especially in those terms that students have anecdotally claimed are overwhelming.*"

Students had only a handful of possible pathways to get through their curriculum in the past. As the department has offered the required courses in both terms, the number of pathways multiplies. Allowing some freshmen to get started early with the curriculum further complicates the pathways. **S1** further commented that "*With CoursePathVis, we can see all these pathways, and we can identify those pathways that have resulted in better outcomes or those that resulted in more problematic ones. This tool will soon become essential in identifying early on which potential pathways some students should not pursue.*"

In terms of improvement, **S1** added that "*CoursePathVis shows the flows for a given graduation year; adding a feature to see the flows for multiple consecutive years could help.*" **S2** commented that "*The tool has allowed us to focus on times when students' grades see drops and prompted a deeper qualitative investigation into the reasons. The COE has used these results to examine if the performance of students from underrepresented minorities, especially students of color, are disproportionately affected by this drop.*" **S3** echoed that "*If the tool could include data about gender, race, and ethnicity, that would allow for another type of exploration into our curriculum's effectiveness.*"

Finally, the three stakeholders agreed that overall, the tool serves as an excellent barometer of the health of the curriculum, advising, and potential curricular changes.

Conclusions and Future Work

We have presented CoursePathVis, a visual analytics tool for investigating how students take courses to enable stakeholders to find patterns, spot outliers and discover insights. The four design requirements are built in the visual interface and interactions, as demonstrated by the results. Three stakeholders joined the design and evaluation of CoursePathVis, and their feedback confirms the usefulness of the tool. In the future, we will address the tool's current limitations and incorporate the stakeholders' suggestions into revision. Besides the required courses, we will include elective courses to provide a holistic picture of the CSE department's course paths. We will apply CoursePathVis to other COE departments and deploy it for student advising and curriculum development. Finally, we will extend this tool to handle other event sequences, such as career paths, tourist paths, and patient treatment records.

Acknowledgments

This research was supported in part by the U.S. National Science Foundation through grants IIS-1455886, CNS-1609394, DUE-1833129, CNS-1855278, IIS-1955395, IIS-2101696, and OAC-2104158. The authors would like to thank the anonymous reviewers for their helpful comments.

References

[1] Advanced placement. https://en.wikipedia.org/wiki/Advanced_Placement

[2] Ali, L., Hatala, M., Gašević, D., Jovanović, J.: A qualitative evaluation of evolution of a learning analytics tool. Computers & Education **58**(1), 470–489 (2012)

[3] Arnold, K.E., Pistilli, M.D.: Course Signals at Purdue: Using learning analytics to increase student success. In: Proceedings of International Conference on Learning Analytics and Knowledge. pp. 267–270 (2012)

[4] Bakharia, A., Dawson, S.: SNAPP: A bird's-eye view of temporal participant interaction. In: Proceedings of International Conference on Learning Analytics and Knowledge. pp. 168–173 (2011)

[5] Bodily, R., Verbert, K.: Trends and issues in student-facing learning analytics reporting systems research. In: Proceedings of International Conference on Learning Analytics and Knowledge. pp. 309–318 (2017)

[6] Charleer, S., Moere, A.V., Klerkx, J., Verbert, K., Laet, T.D.: Learning analytics dashboards to support adviser-student dialogue. IEEE Transactions on Learning Technologies **11**(3), 389–399 (2017)

[7] Chen, Q., Yue, X., Plantaz, X., Chen, Y., Shi, C., Pong, T.C., Qu, H.: ViSeq: Visual analytics of learning sequence in massive open online courses. IEEE Transactions on Visualization and Computer Graphics **26**(3), 1622–1636 (2020)

[8] Deng, H., Wang, X., Guo, Z., Decker, A., Duan, X., Wang, C., Ambrose, G.A., Abbott, K.: PerformanceVis: Visual analytics of student performance data from an introductory chemistry course. Visual Informatics **3**(4), 166–176 (2019)

[9] Fu, S., Zhao, J., Cui, W., Qu, H.: Visual analysis of MOOC forums with iForum. IEEE Transactions on Visualization and Computer Graphics **23**(1), 201–210 (2017)

[10] Gotz, D., Stavropoulos, H.: DecisionFlow: Visual analytics for high-dimensional temporal event sequence data. IEEE Transactions on Visualization and Computer Graphics **20**(12), 1783–1792 (2014)

[11] Goulden, M.C., Gronda, E., Yang, Y., Zhang, Z., Tao, J., Wang, C., Duan, X., Ambrose, G.A., Abbott, K., Miller, P.: CCVis: Visual analytics of student online learning behaviors using course clickstream data. In: Proceedings of IS&T Conference on Visualization and Data Analysis. pp. 681–1–681–12 (2019)

[12] Guo, S., Xu, K., Zhao, R., Gotz, D., Zha, H., Cao, N.: EventThread: Visual summarization and stage analysis of event sequence data. IEEE Transactions on Visualization and Computer Graphics **24**(1), 56–65 (2018)

[13] Guo, Y., Guo, S., Jin, Z., Kaul, S., Gotz, D., Cao, N.: A survey on visual analysis of event sequence data. IEEE Transactions on Visualization and Computer Graphics, (2021)

[14] Heileman, G.L., Babbitt, T.H., Abdallah, C.T.: Visualizing student flows: Busting myths about student movement and success. Change: The Magazine of Higher Learning **47**(3), 30–39 (2015)

[15] Horváth, D.M., Molontay, R., Szabó, M.: Visualizing student flows to track retention and graduation rates. In: Proceedings of International Conference on Information Visualisation. pp. 338–343 (2018)

[16] Jivet, I., Scheffel, M., Specht, M., Drachsler, H.: License to evaluate: Preparing learning analytics dashboards for educational practice. In: Proceedings of International Conference on Learning Analytics and Knowledge. pp. 31–40 (2018)

[17] Krumm, A.E., Waddington, R.J., Teasley, S.D., Lonn, S.: A learning management system-based early warning system for academic advising in undergraduate engineering. In: Learning Analytics, pp. 103–119. Springer (2014)

[18] Lim, L., Dawson, S., Joksimovic, S., Gašević, D.: Exploring students' sensemaking of learning analytics dashboards: Does frame of reference make a difference? In: Proceedings of International Conference on Learning Analytics and Knowledge. pp. 250–259 (2019)

[19] Lonn, S., Aguilar, S., Teasley, S.D.: Issues, challenges, and lessons learned when scaling up a learning analytics intervention. In: Proceedings of International Conference on Learning Analytics and Knowledge. pp. 235–239 (2013)

[20] Millecamp, M., Gutiérrez, F., Charleer, S., Verbert, K., De Laet, T.: A qualitative evaluation of a learning dashboard to support advisor-student dialogues. In: Proceedings of International Conference on Learning Analytics and Knowledge. pp. 56–60 (2018)

[21] Raji, M., Duggan, J., DeCotes, B., Huang, J., Zanden, B.V.: Visual progression analysis of student records data. In: Proceedings of IEEE Symposium on Visualization in Data Science. pp. 31–38 (2017)

[22] Santos, J.L., Verbert, K., Govaerts, S., Duval, E.: Addressing learner issues with StepUp!: An evaluation. In: Proceedings of International Conference on Learning Analytics and Knowledge. pp. 14–22 (2013)

[23] Trimm, D., Rheingans, P., desJardins, M.: Visualizing student histories using clustering and composition. IEEE Transactions on Visualization and Computer Graphics **18**(12), 2809–2818 (2012)

[24] Verbert, K., Ochoa, X., De Croon, R., Dourado, R.A., De Laet, T.: Learning analytics dashboards: The past, the present and the future. In: Proceedings of International Conference on Learning Analytics and Knowledge. pp. 35–40 (2020)

[25] Wang, T.D., Plaisant, C., Quinn, A.J., Stanchak, R., Murphy, S., Shneiderman, B.: Aligning temporal data by sentinel events: discovering patterns in electronic health records. In Proceedings of ACM Conference on Human Factors in Computing Systems. pp. 457–466 (2008)

[26] Wongsuphasawat, K., Gotz, D.: Exploring flow, factors, and outcomes of temporal event sequences with the Outflow visualization. IEEE Transactions on Visualization and Computer Graphics **18**(12), 2659–2668 (2012)

[27] Wu, T., Yao, Y., Duan, Y., Fan, X., Qu, H.: NetworkSeer: Visual

analysis for social network in MOOCs. In: Proceedings of IEEE Pacific Visualization Symposium. pp. 194–198 (2016)

[28] Zhang, Z., Gotz, D., Perer, A.: Iterative cohort analysis and exploration. Information Visualization, **14**(4), 289–307 (2015).

Author Biography

Brendan J. O'Handley *is an M.S. student in the Department of Computer Science and Engineering and an application development specialist in the Office of the Registrar at the University of Notre Dame. His main research interests are visualization in education and learning analytics.*

Morgan K. Ludwig *received her B.S. degree in computer science from the University of Notre Dame in 2020. Currently, she is a software development engineer at Amazon Web Services.*

Samantha R. Allison *is an undergraduate student in the Department of Computer Science and Engineering at the University of Notre Dame.*

Michael T. Niemier *is a professor in the Department of Computer Science and Engineering at the University of Notre Dame. He received a Ph.D. degree in computer engineering from the University of Notre Dame in 2004. His research interests include designing, facilitating, benchmarking, and evaluating circuits and architectures based on emerging technologies.*

Shreya Kumar *is an assistant teaching professor in the Department of Computer Science and Engineering at the University of Notre Dame. She received a Ph.D. degree in computer science from Michigan Technological University in 2016. Her research interests areas are software engineering, human computer interaction, gerontechnology, digital literacy initiatives, integrating ethics and societal impact driven focus into the computing curriculum and recruitment of women in computing.*

Ramzi Bualuan *is a teaching professor in the Department of Computer Science and Engineering at the University of Notre Dame. He received an M.S. degree in electrical engineering from the University of Notre Dame in 1985. His research emphasizes databases.*

Chaoli Wang *is an associate professor in the Department of Computer Science and Engineering at the University of Notre Dame. He received a Ph.D. degree in computer and information science from The Ohio State University in 2006. Dr. Wang's main research interests are scientific visualization and visual analytics.*