1. Meeting Date, Scribe=Name

1.1. Questions/points to bring up:

DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

(1) Add questions and/or thoughts that you want to discuss in the meeting here.

1.2. Meeting Summary:

DG: This subsection is the scribe's responsibility.

- Who was in attendance?
- What was discussed?

  DG: Be sure to mention everyone's ideas regardless of whether they seem to work out or not. Give a brief description of the attempt, and reasons why and why not the group found it to be promising. (Ofc, try to stay nice and objective)
- What are the goals for the next meeting?

  DG: One can think of this as a sort of to-do list for the next meeting.

2. Monday, Sept. 29, Scribe=Dan

2.1. Questions/points to bring up:

2.1.1. *Logistics and General Questions:*

(1) DG: Will people be divided into two groups for the two aspects of the project or will they be able to bounce around?
(2) DG: What kinds of real-world examples do you have of convex vs. concave cost functions?
(3) DG: Will our group interact much with the other WXML groups throughout the quarter?
(4) DG: Will the matching algorithm that we design for the math TA's necessarily have a concave cost function?
(5) DG: When will our weekly meetings happen?

2.1.2. *Technical questions/comments from Ottolini-Steinerberger [***ottolini˙2025***].*

(1) DG: I think that I understand how one performs the Dyck matching, but can we rigorously explain this algorithm?
(2) DG: Consider the statement of Theorem 1. It does not say that the Greedy matching is just a constant away from the optimal. Instead, it says that the Greedy matching is a constant away from the estimate given by Hölder's inequality. However, in Hölder's inequality, we have an extra factor depending on n. How should one intuitively understand this extra factor? Is it the price one must pay for approaching any point set with a single algorithmic tool?
(3) DG: On page 6, consider the paragraph before Corollary 2. You give an "interesting heuristic". I do not understand this heuristic. Can we talk more about how it works?
(4) DG: Consider Figure 7. It makes sense that greedy gets better when you have low values of p. Do you have any explanation as to why the Dyck matching is not monotonic? That is, "why is there hill in the Dyck graph?"

(5) DG: On the bottom of page 8, you reference a celebrated theorem of Birkhoff and Von Neumann in the area of optimal transport. Can you give some insight into their proof methods and why one should expect (or not) this result to be true?
(6) DG: Consider the fourth centered equation on page 10. How is one achieving this bound for the integral? Is it a (Left or Right) Riemann estimate?
(7) DG: Consider the result of Ajtai-Komlos-Tusnady cited on the bottom of page 10. This feels very out of the blue. Again, can you provide any insight into this result? Why should one even expect this sum to be on the order of $\sqrt{n \log(n)}$?

2.1.3. *Questions/comments from Wikipedia.*

(1) DG: In Section "Different Stable Matchings" of "Stable Matching Problem" notice that there are two stable matchings that seem quite unfair. Namely, that either men get their first choice or women get their first choice. So, it very much matters whose side you want to favor when creating a stable matching via Gale-Shapley (in which case, it is the proposing side, i.e. men, that are favored). This is to say, plz favor the TAs when building the matching machine, haha.
(2) DG: The Wikipedia mentions that there can be many stable matchings, and that they carry the structure of a finite distributive lattice. Is there some variation of Gale-Shapley that one can do to recover every stable matching? Idk, something like, midway through running Gale-Shapley, ask (all or some subset of) the women to propose? ¯_(ツ)_/¯

2.2. **Meeting Summary:**

- Who was in attendance?
  - Lufan, Zareef, Jasper, Alan, Jonathan, Stefan, Dan.
- What was discussed?
  (1) Stefan will be giving a crash-course of how to apply to grad school on Wed from 4pm-6pm in DEN 110.
  (2) There should be freedom in this project! That is, students should bounce between the two aspects of the project.
  (3) The Plan for the Algorithm:
    (a) Spend 2 weeks developing "What is the important information that I want or need to input into the solver?"
    (b) After 2 weeks, we meet with Sarah Garner (director of student services) and ask her all of the questions that we can possibly dream of.
    (c) This project will have to deal with questions like "What if a TA selects 3 classes that they really want to teach and say not interested to the others?" Should we reward them or reward someone who is more of a team player and gives several options?
    (d) Note: One approach could be to write some minimal raw code that takes in some input, and then gives an output that has some randomness to it. Sarah could then choose her favorite of these options.
  (4) The plan for the theoretical project:

(a) There has been much work done for cost functions with $p \geq 1$. This is an incredibly well-studied area of research in optimal transport. However, things are cursed when $p < 1$. Like, in this world, the triangle inequality doesn't even work. Yet, the $p < 1$ world is the world that we live in. This is the world of *concave* cost functions.
(b) The good news is that Ottolini-Steinerberger [**ottolini˙2025**] gives some insight into this world. Namely, for p very small, the greedy algorithm performs quite well. Meanwhile, for p close to 1, the Dyck matching seems to perform quite well.
(c) One potential avenue to begin with is the following: consider $p = \frac{1}{2}$. In this case, it seems that neither greedy nor Dyck is spectacular. Can we come up with a method that works for these "middle" values of p?
(d) Another idea that Jonathan brought up is the following.
    (i) Consider the problem of n houses and n pizzerias all on a line. Now, as Stefan noted, the best way to pair up the pizzerias and the houses is to simply match things left to write when the cost function has $p \geq 1$.
    (ii) Jonathan's question is "In the context of this toy example of pizzerias and houses, does this strategy of pairing pizzerias and houses still work when $p < 1$?" No one has an answer, so we need to test for it.

- What are the goals (or things to try) for the next meeting? DG: Ofc, no need to do everything. Choose your favorite thing to do from the buffet.
  - Stefan:
    * Send us some working code.
    * Write Sarah Garner and ask for a sample TA form.
  - Dan:
    * Schedule meeting times.
    * Get on Discord.
  - Everyone else:
    (1) Write down as many questions that you can possibly think of regarding the kind of inputs that you want to go into the algorithm.
    (2) Try to test Jonathan's question of "Does Greedy work for houses and pizzerias on a line when $p < 1$?"
    (3) Read Ottolini-Steinerberger [**ottolini˙2025**]. There is NO need to read about Corollary 1, Theorem 2, Theorem 3, Theorem 4, Proposition 2, or Corollary 2.
    (4) Can we think of possible optimal strategies for matching point sets with $p \approx 1/2$?
    (5) Please update the Overleaf with any progress or ideas!!
    (6) Some working code from Ottolini-Steinerberger. Google Colab Notebook

3. Friday Oct. 3, Scribe=Dan

3.1. Questions/points to bring up:

(1) ZA: Some working code from Ottolini-Steinerberger. Google Colab Notebook

3.2. **Meeting Summary:** DG: This subsection is the scribe's responsibility.

- Who was in attendance?
  - Stefan, Dan, Zareef, Jonathan, Lufan, Samarth, Jasper.
- What was discussed?
  - With the right definition, Jonathan gave a nice interpretation of the Dyck matching by using parentheses. Roughly, speaking, wlog, assume that the first point on the line is red. Then, for all red points, put "(" and for all blue points put ")". Afterwards, match up the parenthetical pairs.
  - Also, Jonathan brought up a good point about why a) the greedy matching is good with p small and b) there is a "hill" in the graph for the Dyck matching in Ottolini-Steinerberger [**ottolini˙2025**]. Namely, the scaling might be off. That is, random matchings might do very good simply because the cost(s) are very small.
  - Zareef got some nice code running to output "circle draw" to capture what the picture of the matchings look like. One can see the code in the shared Google Colab Notebook.
  - Stefan asks if we can do better with a 2-step or a k-step greedy algorithm. That is, at a given step, ask "what is the best collection of k pairs that I can pull from my set before resetting for the next step?". Notice that $k = 1$ is the usual greedy algorithm while $k = n$ is the optimal matching.
  - Stefan made some more observations:
    - For p small, he and Andrea observed that the first pair matched with the greedy algorithm was in the optimal matching roughly 98% of the time. Meanwhile the last pair that the greedy matching made was overwhelmingly not in the optimal matching.
    - Suppose two points are very very very close together, like, stuck to each other like two lovebirds holding hands on a sunset walk around Green Lake. Like, suppose even that a red and a blue point are stacked on top of one another. Must they always be paired in an optimal matching?
  - Dan made the following suggestion.
    - We know that Greedy is good for p very close to 0. We know that Dyck is good for p very close to 1. For p roughly in the middle, something like $p \approx 1/2$, we can do a hybrid algorithm that starts by doing greedy, but then switches to Dyck midway through?
- What are the goals for the next meeting? DG: Again, choose your favorite(s) from the buffet
  (1) NEW:
    (a) Recompute the performance of the greedy matching, but now compared to random matchings.
    (b) Answer Stefan's question of "Suppose even that a red and a blue point are stacked on top of one another. Must they always be paired in an optimal matching?"

(c) Run experiments on Dan's Frankenstein idea of doing Greedy first then switch to Dyck. If it performing better than either strategy alone, when is the best time to switch? How should the optimal time to switch depend on p?

(2) OLD:
   (a) Write down as many questions that you can possibly think of regarding the kind of inputs that you want to go into the TA algorithm.
   (b) Read Ottolini-Steinerberger [**ottolini˙2025**]. There is NO need to read about Corollary 1, Theorem 2, Theorem 3, Theorem 4, Proposition 2, or Corollary 2.

4. Tuesday Oct 7, Scribe=Zareef

4.1. **Questions/points to bring up:** DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

(1) ZA: We learned that the optimal algorithm for $p \geq 1$ is matching in ascending order the i-th largest x_i to y_i. How well does this algorithm work for $p < 1$? How does it vary as p ranges?
(2) ZA: What is a good way to measure the efficacy of a model?... cost function, distribution of x_i, y_i

4.2. **Meeting Summary:** DG: This subsection is the scribe's responsibility.

- Who was in attendance? Lufan, Jasper, Alan, Zareef, Jonathan, Samarth
- What was discussed? DG: Be sure to mention everyone's ideas regardless of whether they seem to work out or not. Give a brief description of the attempt, and reasons why and why not the group found it to be promising. (Ofc, try to stay nice and objective)
  - Samarth's proof of the triangle inequality. Because $x^{\frac{1}{p}}$ is monotonic it holds. SV: It's also true for $0 < p < 1$ in general but the triangle inequality we don't have is Minkowski's Inequality, which is reversed in this range. See more here
- What are the goals for the next meeting?
  - When doing TA results, TA discussions the form should request TA's to look at the time schedule time schedule should be released to them first.
  - There are three parts of creating an algorithm. Getting the data, choice of data, and outputting an algorithm. Implementation of the algorithm should be towards tree/pruning. The steps of the pruning be professors matching, then 1st and 2nd years.
  - For TA, should they choose blocks for their time availability. Does a website makes sense to encode UW information and schedule (much like the schedule app). One concern about having all the data on a website is a privacy reason. Should we ask whether or not a website is okay?
  - Historically have there been more 1st and 2nd year students than sections?
  - What is the way that Sarah is making decisions about time conflicts?
  - What are we expected to do with "Maybe" TA's?

- What is methodology for manually doing it?
- For an algorithm on scheduler start with the people that are only available for one time slot and then remove them. We can flag potential problems where there are holes in schedules. Is it possible to get the time schedules? What is the format you want to receive?
- The algorithm is split between 12x and 1st/2nd year grad students. One method for implementation is Gale Shapley's with indifference so courses have preference for 1st years and 2nd years. What is the balance between getting information for time conflicts and putting people based on judgement of their time conflict.
- Jonathan was working on a graph that showed the specific region where greedy matches.

DG: One can think of this as a sort of to-do list for the next meeting.

- When are we interviewing Sarah, and what are our top questions? In what order? SV: My top questions for Sarah are a) how does she manually create assignments? (including deciding about time conflicts) b) Have their been more 1st/2nd years than sections and if so what happens to the excess students? c) what data can we actually have access to both from students and logistically (like section times). If we can ascertain exactly what data we will have access to, then we may be able to create a toy data set with that data in an easy-to-use format (csv?) or get previous quarter data and then attempt to implement algorithms on the toy data. We could also rewrite/update our existing toy data. JDS: I'd also like to ask what's the best type of interface for our app, both on the input and output side: does she want to be able to mark students unavailable for certain times, or assigned to certain sessions, manually, or should it be automatic? Does she want one possible matching, or multiple? How are the inputs going to be presented, both for section (times) and for TAs? Does she want, say, the functionality to look at a possible matching, tell the computer what was *wrong* about it, and get an updated version?ZA: Is giving grad students a TA assignment preferential to another student getting a lower choice? For example, if a TA is not very available then would they be given a TA assignment that another TA sees as high preference?

5. Friday Oct 10, Scribe=Samarth

5.1. **Questions/points to bring up:** DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

(1) Nothing

5.2. **Meeting Summary:** DG: This subsection is the scribe's responsibility.

- Attendance: Samarth, Jasper, Alan, Jon, Lufan, Dan, Stefan
- Jasper shared his combination of Greedy + Dyck.
  (1) Suspects that the Greedy is consuming good Dyke matches.
  (2) Will rerun with more points to circumvent potential "small n problem" (the paper uses $n = 250$, for no particular reason)
- Stefan says good exercise to run Greedy on 90% of generated points and check remaining ones for patterns

(1) Could be related to Brownian motion
(2) Hypothesis: long transport point matching occurs between subsets that have "too many" points of that type generated
(3) Run for different n and p as well and see if you can spot anything

6. Tuesday Oct 14, Scribe=Alan

6.1. **Questions/points to bring up:** DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

(1) Nothing

6.2. **Meeting Summary:** DG: This subsection is the scribe's responsibility.

- Attendance: Lufan, Samarth, Alan, Jon, Dan, Zareef
- What was discussed?
  - Samarth suggested running Gale Shapley to match students with classes first, then matching these subsets to specific times for sections.
  - Alan suggested initially processing specific requests from TAs to work with professors and then working downstream based on seniority for those with special preferences to specific classes.
  - Jon presented a random data setting in a notebook where we can simply "optimize" without running any Gale Shapley. The weights of each point are determined
  - Jon explained how a system involving front-end interfaces with interactivity allows us to improve the performance of the algorithm and setup.
  - Samarth explained how working on the algorithm is most important.
  - Alan asked about the possibility of notifying Sarah whenever someone utilized the "any additional information" text box on the TA Google Form, so she could get in contact with professors and get their approval on special requests that are shared.
  - Dan suggested that since course requests required professor input as well, we should instead send out another form to the professors for them to share what TA they would like to work with specifically and for what course.
  - Jon shared how the Google Form's data collection duration influences the results of our data collection.
  - Dan shared the timeline for the Google forms, and the classes are assigned a month after the collection deadline.
  - Jon ran SDP on data and gave us an initial result that had no conflicts.
- What should we do next before Friday's meeting?
  (1) Incorporating "the rules" and adding consideration for time data into the algorithm.
  (2) Consider k-greedy algorithm
  (3) Think about what kind of front end we want to design or work with for the TA algorithm.
  (4) What data do we want to collect?
  (5) Further investigate Greedy vs. Dyck, while also double checking and cleaning the existing code. How can we improve efficiency?

(6) How are things different between the SDP setup and Gale Shapley (if we can build an algorithm using GS with indifference)?
(7) Jon: write email to Stefan about spike.
(8) **Reminder:** This Friday's (10/17) meeting will be at Dan's office at our normal meeting time for Fridays.

7. Friday, Oct. 17, Scribe=Rebecca

7.1. **Questions/points to bring up:** DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

(1) Nothing

- Attendance: Lufan, Samarth, Alan, Jon, Dan, Jasper
- What was discussed?
  - The group explored optimization strategies for concave cost matching, focusing on how point distribution and local-improvement algorithms influence global optimality.
  - Alan described generating large-scale randomized datasets (up to 1000 points per setup) to analyze how the spatial structure of point clouds, such as density, clustering, and spread, affects performance. He proposed comparing dense and sparse cases, computing average and minimum distances between points, and using statistical thresholds to determine when to switch between Greedy and Dyck matchings. He also suggested experimenting with multidimensional and subset-based variations to evaluate when switching occurs near boundary conditions.
  - Dan suggested starting with a Greedy matching, then testing whether local swaps can improve total cost, effectively creating a local-improvement heuristic. He also proposed comparing results of Greedy vs. Dyck matchings when the blue points have different spread patterns, observing how distance distribution influences efficiency.
  - Samarth focused on testing Greedy matchings instead of random ones to build a comparison baseline. He proposed analyzing when pair distances (n^2 comparisons) start dominating the cost and determining how many local swaps are required to reach optimality. He also noted that even a "simplest local best" method might still show linear growth in performance.
  - Jonathan discussed integrating semidefinite programming (SDP) and constraint weighting into the TA assignment model. He explained assigning strong penalties or bonuses (e.g., +1000) to enforce hard preferences, such as guaranteeing first-year TA placements in 12x courses. He highlighted that one of the biggest challenges remains converting the spreadsheet data, particularly time conflicts and TA preferences, into a machine-readable format suitable for optimization.
  - We agreed that current issues primarily stem from unstructured input forms. Jonathan noted encoding problems in CSVs (e.g., mixed data types, malformed UTF-8). The solution discussed was building a standardized preprocessing pipeline to clean, normalize, and align TA and course data (columns: name, course, time, preference level).
  - Alan and Jasper discussed interface improvements, proposing separate Google Forms for TAs and professors. The TA form would collect

structured data (availability, course preference, conflicts), while the professor form would include fields for TA name and preferred pairing. Jasper suggested using "yes/no" dropdowns per course and a textbox only for special cases. This eliminates free-text ambiguity and reduces Sarah's manual editing workload.
  - We emphasized prioritizing backend optimization (algorithm logic) rather than polished frontend design. The final system should allow partial manual intervention, Sarah can lock certain assignments, and the algorithm reruns optimization on the remaining pool.
- What should we do next before Friday's meeting?
  (1) Generate datasets with varying density and dimensionality to test how point structure affects algorithm performance.
  (2) Implement and test the local-swap improvement heuristic; measure convergence rate and whether it reaches the global optimum.
  (3) Define a threshold rule (average vs. minimum distance) to trigger switching from Greedy to Dyck matching.
  (4) Build a preprocessing script that standardizes TA and section spreadsheets (clean headers, normalize preferences, resolve encoding).
  (5) Design two structured Google Forms, one for TAs, one for professors, and simulate merged data for matching.
  (6) Experiment with SDP formulations enforcing strong preference constraints (+1000 weights).
  (7) Prepare visual comparisons for local vs. global matchings across randomized and structured point distributions.

8. Tuesday, Oct. 21, Scribe=Zareef

8.1. **Questions/points to bring up:** DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

(1) Nothing

- Attendance: Lufan, Samarth, Alan, Jon, Dan, Zareef, Jasper
- What was discussed?
  - Jonathan's code does preference for courses and we need preferences for time. Preferences for time is easy, but time conflicts are hard.
  - How can we directly take in an excel/google sheets file and clean the time data.
  - There is a problem with GSR's and the number of sections taught. We (Sarah) can change the number of sections taught so that it matches.
  - We will also need another spreadsheet that has professors who are directly assigned to which sections.
  - Jasper said he was thinking about the geometry of optimal matchings (in higher dimensions)
  - Jasper thinks that hybrid matching algorithm of greedy and dyck should be non crossing. We have an idea that the proof works.
- What should we do next before Friday's meeting?
  (1) Dan should email Sarah to ask for the time schedule (in excel format) of Winter Quarter.
  (2) Zareef and Alan should clean the data so Jonathan can read the combined sections as one section. Make sure that SDP is ready next week.

(3) If Jasper and Samarth wants to work on changing distributions of matchings for k-greedy.
(4) Jasper and Samarth can work on the formalization of the lemma that hybrid matching algorithm is non crossing.

9. Friday Oct 24, Scribe=Samarth

9.1. **Questions/points to bring up:** DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

(1) JB: Jasper
   (a) (Progress) Non-crossing of McCann circles for Greedy-Dyck hybrid algorithm in concave regime proof
   (b) (Progress) k-swap greedy algorithm empirical results (good)
   (c) (Question) CS friend proposed "Just turn it into a convex problem". Is there any merit to this?
   (d) (Question) New approach proposal (maybe computationally unreasonable): find some generalized property to generate all permutations of matchings where McCann circles are non-crossing. Then search that space for optimal (semi brute force).
   (e) (Question, related to above) Maybe there are other ways to characterize an optimal solution which will whittle down our search even further?

9.2. **Meeting Summary:** DG: This subsection is the scribe's responsibility.

- Attendance: Lufan, Samarth, Jasper, Alan, Jon, Dan, Stefan, Zareef
- Jasper shared his proof of the Greedy-Dyck hybrid
  (1) Not just a hack since the geometric structures of the circles are preserved
  (2) Stefan wants to see how it changes depending on when we switch from Greedy to Dyck
- Jasper shared code that does Greedy then k-swaps
  (1) In Jasper's observations, it appears to converge to optimal
  (2) We hypothesize that it should always converge
- TA Machine
  (1) Dan asked Sarah to send winter time schedule data to come up with matches for winter
  (2) Alan shared the code that takes original Excel data and cleans it up for example by merging back-to-back section
  (3) Stefan has a proposal for measuring TA happiness – human proposal, not math porposal!
     (a) Suppose TA A gives a balanced preference list and TA B is picky and wants a specific class, and TA C says nearly everything aside from 1-2 classes
     (b) Stefan says we should do whatever we can to not give C the one they want and and to not care about B because they are too picky. To do this, subtract mean and devide by variance each person's preferences (C transform)

10. Tuesday Oct 28, Scribe=Jon and Jasper

10.1. **Questions/points to bring up:** DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

10.2. **Meeting Summary:** DG: This subsection is the scribe's responsibility.

- Attendance: Lufan, Samarth, Jasper, Alan, Jon, Dan, Zareef
- Dan is giving a talk Friday, Nov. 7th at 2:30-3:30 in PDL C-38 about his work, "come if you want."
- TA Machine:
  - Discussed the time format for TAs/courses.
  - We have some of the new data for Winter from Sarah, but we're gonna wait until we have all the TA responses in.
  - Idea from Dan: since most of the time conflicts were from seminars, a future time request form should include them as options directly. (Cut down on human involvement in making text input machine readable.)
- Optimal Transport:
  - Can we prove that "non-crossing" matches can be arbitrarily bad?
    - Consider a set of points such that first rbbr are each ϵ away from each other and N away from the next points brrb, also ϵ away from each other.
      The optimal matching cost incurred is $4\epsilon^p$ in the trivial case of matching close points. In contrast, we can construct matchings incurring cost $> N^p$ by matching in a "rainbow" pattern which is non-crossing.
    - idea: for any two points we can push them out towards 0, and 1 respectively, their distance being N.
  - Can we write an upper bound for the cost of a hybrid algorithm in terms of the cost of greedy and Dyck on the same pair of points.
  - Can total cost of Dyck of a subset be worse than total cost of Dyck on the whole set?
    - Proof simple by counterexample using something similar to what wass described in the arbitrarily bad non-crossing matching.
    - take *rbbr* a distance N away from *brrb*. Dyck matching the whole set costs us $\approx 4\epsilon$. If we take the subset of the points containing the outer 3 points on either side, now we have a dyck cost of $\approx 2\epsilon + N > 4\epsilon$, where N is large (at least greater than 2ϵ).
  - Where is the crossoever point of k where greedy gets worse?
  - Can we find some bound for Hybrid(k), like k^n greedy $1 - k^n$ Dyck? Or to find some k^* that tends to perform the best, or some bound for what k^* should be, even if we don't know its value? How does all this depend on p or on n?
  - Can we show that any Hybrid is better than (total of Greedy) + (total of Dyck) for general k?
  - To make it precise: "Can we find an interesting bound of Hybrid(k) in terms of $Greedy_{opt}, Dyck_{opt}, n, p$ (with n of each color of points and p the power) and *not* the distribution of points?"
  - Note: we can force Dyck to be arbitrarily bad, but our examples for that make Greedy do better.

  - Computational idea: find k^* for a bunch of different sets of points.
  - Idea: Consider $D = \{[d(red_i, blue_j)]^p\}_{i,j\in[n]}$. Conjecture: the best time to stop greedy (i.e. k) is when the distance of next pair we would match is $>$ the mean of D.
- TA Machine:
  - Discussed the

11. Friday Oct 31, Scribe=Alan

11.1. **Questions/points to bring up:** DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

11.2. **Meeting Summary:** DG: This subsection is the scribe's responsibility.

- Attendance: Lufan, Alan, Dan, Jon, Samarth, and Jasper
- TA Machine:
  - Jon showed the data visualizations created from our current iteration of the sorting algorithm,
  - Alan suggested using a parallel sort rather than sequential sorting in order to process the difference between "course" and "time" preference.
  - Stefan suggested utilizing a for loop to test possible weights for our algorithm and storing all weights that are "viable" so that we have a database of weights that can be utilized down the line in the future.
  - Stefan wants us to share our current results with Sarah and get some ideas from her on what to change with our current setup.
  - Zareef is checking how Sarah's matchings will perform in our algorithm.
  - In the coming weeks, we will be looking into how to format inputs and outputs for our algorithm so that it can be reproducible and reusable for various applications in the future.
- Optimal Transport:
  - Jon and Jasper discussed a hypothesis about possible matches. Jon explained how the structure of Dyck matching could allow a possible subset selection in which taking elements that are part of the slopes and replacing them with Greedy matches instead of Dyck ones, which could still potentially keep the peaks (local optimums) in the long run.
  - Stefan suggested possibly plotting the Greedy matches using the Dyck setup.
  - Jasper shared his findings of using the hybrid Greedy/Dyck match, and the next step is to chain the various images together into a "movie" so that we can see the changes as there are changes in n and p.
  - No proofs on the cost of the hybrid method currently. We need to figure out whether or not such hybrid matches has a cost bound (that we can figure out). Is it possible for the cost to be bounded in relation to Greedy/Dyck, or is it just some random values that we can hypothesize over time?
  - We know that Dyck is good when $p \approx 1$, so we need to figure out when to switch over to the Greedy matches.

  - Dan suggested a possibility: if we were to utilize the hybrid method after one, two, etc. matches, would the optimal matching always be one of the hybrid options?
  - Stefan highly recommended utilizing visualizations to discern patterns in our work and exploration.
  - Stefan suggested using gradient descent in higher dimensions to move specific points around and explore how such movements will change the costs and patterns.
- TODO: Let Zareef finish up his work, then we schedule a meeting with Sarah to get her feedback on the matching algorithm.

12. Tuesday Nov 3, Scribe=Jasper

12.1. **Questions/points to bring up:** DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

12.2. **Meeting Summary:** DG: This subsection is the scribe's responsibility.

- Attendance: Lufan, Alan, Dan, Jon, Samarth, and Jasper
- TA Machine:
  - Ask Sarah what is up with the repeated MATH 208 D
  - Zareef finished his cleaning of last quarter's data, which had many weird edge cases (such as MATH 207 in Italy or the repeated MATH 208's)
  - We also saw that Sarah's matching for Fall breakdown to 44/18/12 prefer/ok/avoid compared to our 54/22/0 (!).
  - Jon is going to finish up cleaning up Winter stuff tonight, and hopefully Dan will be able to send Sarah an email tomorrow morning which we can discuss on Friday.
- Optimal Transport:
  - Jon made a nice video
  - Can we find a case where dyck is optimal under strict concave costs?
  - Can we use a genetic algorithm to generate a set of points such that the hybrid algorithm is worse than either greedy or Dyck for all values of k?
  - Take $2n$ points distributed equidistant, paint them randomly. Find heuristics for what makes for good dyck vs greedy matchings.
  - see if hybrid is better for uniformly distributed points (10,000+ simulations) via brute force.

13. Friday Nov 7, Scribe=Alan

13.1. **Questions/points to bring up:** DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

13.2. **Meeting Summary:** DG: This subsection is the scribe's responsibility.

- Attendance: Lufan, Alan, Dan, Jon, Samarth, and Jasper, Stefan
- TA Machine:
  - Dan shared our results since Tuesday, which (critically) includes the feedback that Sarah gave us. We discussed possibilities of extending the project this quarter and also next quarter.

  - Jon slightly tweaked the TA weighting algorithm, specifically the weighting of the "avoid" cases, which allowed us to further optimize the algorithm's assignments for the upcoming winter quarter.
  - Jon mentioned that a problem was having more ASEs in the winter quarter data than sections available. This is an issue we need to follow up by asking Sarah for feedback and how to proceed.
  - Stefan asked us to propose questions for Sarah regarding how much we favor fourth years versus other years. He believes that fourth year grad students rather than sixth years should get their favorite choices.
  - Things to ask Sarah:
    * Check TA - Section - ID
    * How much to favor older students? Maybe $435^{+}21$? Check with Sarah and also ask her how to scale these values.
- Optimal Transport:
  - Jasper shared the interactive visualization we created for the mountain picture as we compare the performance of hybrid/greedy/dyck.
  - A big question is whether Greedy and Dyck are always optimal for certain situations. Stefan believes there is specific endpoint behavior despite the dips we observe in Jasper's plots.
  - Stefan suggested creating a new plot by running a 1000 iteration for loop through which we would transform all the plots into a 0 and 1 range and explore the concavity of the plot in these various situations. His focus is on running large-scale experiments with lots of points so that we can draw more conclusions utilizing these large numbers of points.
  - Alan suggested a possible bias-variance tradeoff between using Greedy only and Dyck only. Samarth also brought up the fact that Greedy should be run for over 50% of the hybrid in order to optimize performance.
  - Jasper shared that there are cases when Greedy is the optimal even with the hybrid setup, whereas there are other times when the hybrid can find an optimal that's even better than Greedy.
  - Stefan highlighted the reasoning why he previously wanted us to shift everything using $\frac{1}{\sqrt{n}}$ because it reflects the Brownian Bridge and random point generation with uniform distributions.

14. Friday Nov 14, Scribe=Jasper

14.1. **Questions/points to bring up:** DG: It is everyone's responsibility to add questions and thoughts in this subsection before each meeting.

14.2. **Meeting Summary:** DG: This subsection is the scribe's responsibility.

- Attendance: Lufan, Alan, Dan, Jon, Samarth, and Jasper, Stefan
- TA Machine:
  - just need Sarah confirmation.
- Optimal Transport:
  - $\phi = \frac{\#\{x_1,\ldots,x_n : x_i \leq x\}}{n}$
  - if partitions with red blue red blue is dyck better than greedy? think —red—blue—red—blue— drop 100, 200, 300 points etc.

- take known length intervals (maybe varying lengths) and take points as $n \to \infty$.
- Stefan's ramblings: $\#greedy \approx (1 - \frac{cp}{\sqrt{n}})n$
- TODO: run greedy until you hit the minimum. Look where the survivors are. How many intervals are there? Stefan things there will probably be very few (not too many, maybe $\sqrt{n}$.
- TODO 2: Take an alternating arrangement of intervals red— blue—red—blue drop in points (uniformly) as $n \to infty$, see if the the structure of optimal matchings converge to the same pattern.
  - start with blue interval is 1/2 length, red interval 1/2 length.
  - try for greedy (should be simple).
  - try for dyck.

15. Tuesday Nov 18, Scribe=Zareef

15.1. Meeting Summary: ZA: This subsection is the scribe's responsibility.

- Attendance: Lufan, Zareef, Dan, Jon, Samarth, and Jasper
- Meeting Notes:
  - Looking at the points that are left after the brownian bridge there are some small cases we can analyze
  - Given the case RB (or BR) with half and half we know that both greedy and dyck give us the same matching. The "x"th point gets matches to the "(1-x)"th point.
  - To formalize this we find the integral as the number of points goes into infinity: $\int_0^{1/2} d_P(x, 1-x)dx$. Note that if you go from nth red to nth blue you get: $\int_0^{1/2} d_P(x, 1/2+x)dx$.
  - We had good progress on the TA assignment, there was one question about TA's not having years (! we should check on this) and there are less TAs that want to teach than sections available (since some responded and didn't want to teach)
  - Sarah asked us the inputting was good, the text input for "list any inputs" there should be a way to categorize i.e. some seminar with some choices.
  - Fine tuning: How to prioritize seniority (z transform for normalization).
- What are the goals for the next meeting
  - We have an interesting matching situation where $S : BRBR$ (or RBRB) where we have: $f_D(p, B_2)$ where D is a Dyck matching, and $f_G(p, B_2)$. We should then be able to say things about f_D and f_G.
  - Is it possible to write a program to see what the unmatched points look at, at any point in the algorithm?

16. Friday Nov 21, Scribe=Alan

16.1. Meeting Summary: ZA: This subsection is the scribe's responsibility.

- Attendance: Lufan, Zareef, Dan, Jon, Samarth, Alan, and Jasper
- Meeting Notes:
  - Jasper shared his findings from the past week. In particular, he noted that there is a relationship between Greedy and Dyck that can be

replicated. In the beginning, the Greedy does its own thing on a microstructure, but the overall macrostructure of the points is still retained.
  - Jasper also shared a second finding: assuming we have partitions as discussed in last week's meeting, both Greedy and Dyck have a closed-form integral solution. Depending on how many partitions we have and their structures, we will have varying solutions.
  - Stefan's suggestion for next steps is to proportion the points within the partitions, and then scale these proportions upward. He hypothesizes that these scaled partitions will have highly similar (if not entirely the same) circles. The intuition is that the circles will be stabilized regardless of number of points, given the proportions are the same.
  - Samarth shared the plot that he generated based on the information we talked about in last week's meeting. It needs to be fit with a specific function to check for results, but it is promising.
  - The main question to answer is in relation to the dip in the cost, and we want to discern how the landscape of the points will show us more patterns and possible information.
  - Jon shared some progress on the TA Machine. Zareef and Alan are working on some important information regarding actionable items to accomplish with regard to the project.
  - Dan shared some concerns about the color scheme of our plots, so we can make it more accessible to those who are color blind. As a result, we came up with a new color scheme to address this issue.
  - Seniority is also a significant issue (generating weights based on that factor). We are still figuring out what to do with the weights in order to optimize the algorithm's performance.
- What are the goals for the next meeting?
  - HI
  - HI

17. Tuesday Nov 25, Scribe=Alan

17.1. **Meeting Summary:** ZA: This subsection is the scribe's responsibility.

- Attendance: Lufan, Dan, Samarth, Jon, Alan
- Meeting Notes:
  - Jon and Alan walked the group through implementation conversations for the next steps the TA matching algorithm will take.
  - Samarth shared a new plot and the findings of the optimal transport problem. We were able to generate a 0.93 R^2 value for the plot Samarth shared, which is very promising.
- What are the goals for the next meeting?
  - Prepare for the next steps and have some results to share with Stefan for next Friday.
  - Alan will try to get the parser specifications done, perhaps with some help from Zareef.

- Everyone, start thinking about how we're going to get our project board set up for the final presentation. Dan will send out some templates for us to consider, and then we can start planning our board out from there.
- Have a great Thanksgiving break!