

# Syllabus

- [Course Information](#)
  - [Course Description](#)
  - [Course Objectives](#)
  - [Course Prerequisites](#)
  - [Meeting Times:](#)
  - [Course Communication](#)
- [Course Resources](#)
  - [Required Materials](#)
  - [Electronic Resources](#)
- [Course Structure and Sequence](#)
  - [Preliminaries](#)
  - [Planning](#)
- [Decision-Making](#)
- [Assignments and Evaluation](#)
  - [Project](#)
  - [Grade Expectations](#)
  - [Grade Assignment](#)
- [Preliminary Schedule](#)
- [Course Policies](#)
  - [Technology Requirements](#)
  - [Course Attendance Policy](#)
  - [Late Work Policy](#)
  - [Accommodations Policy](#)
  - [COVID 19 Precautions](#)
  - [Academic Integrity](#)

## Course Information

Title: Special Topics – Planning and Decision-Making

Number: MECH 5390 / 6390

CRN: 19308 / 16093

Term: Fall 2022

Instructor: John Bird

Office: Engineering Building A-115

Instructor email: [jjbird@utep.edu](mailto:jjbird@utep.edu)

Instructor phone: 915.747.8406

Office Hours:\

## Course Description

Planning for robotic systems is concerned with generating the sequence of inputs that takes a system from a starting configuration to a goal configuration. This challenge is fundamental to mobile robotics, but also to manipulation and transportation system design. Many planning problems are also closely related to decision-making, where an agent must choose from a set of options to achieve some objective. In this course we will study representations of the planning environment, mathematical tools used to plan robotic motion, and canonical planning algorithms. We will also study planning for systems subject to uncertainty, and connections between planning and decision-making for robotics.

## Course Objectives

At the conclusion of this course you will be able to:

- Identify appropriate representations for an environment
- Describe tradeoffs for various planning approaches in a given problem
- Identify key information in technical papers
- Implement planning algorithms from technical descriptions
- Identify failure modes and inefficiencies in a planner
- Implement basic decision-making algorithms

## Course Prerequisites

This course has no formal prerequisites but a basic knowledge of python programming, differential equations, and linear algebra is important. We will be using python extensively in this course as it is widely used in robotics. You need not have any understanding of python at the outset, but you should have a firm foundation in writing functions in MATLAB, python, or another language.

a couple of guides to python for those who have used matlab

- <https://numpy.org/doc/stable/user/numpy-for-matlab-users.html>
- <https://hprc.tamu.edu/files/training/2018/Spring/Python4MATLAB.pdf>
- <https://docs.python.org/3/tutorial/index.html>

## Meeting Times:

Class will meet from 4:30 to 5:50 pm Monday and Wednesday in room 122 in the Liberal Arts building.

## Course Communication

Office Hours:

tbd or by appointment

Email is the best way to contact me. I will attempt to respond within 24-48 hours, please include the

course title or number in your email subject.

## Course Resources

### Required Materials

No textbooks are required for this class, but the content will draw heavily on:

- [Planning Algorithms by Steven LaValle](#)
- various papers on planning

Homework will not be assigned from the text and it is available online so there is no specific need to purchase a textbook or a particular version. However, if you plan to continue working in this area having a reference text is very handy.

### Electronic Resources

Modern engineering practice draws heavily on electronic resources broadly available including technical papers, content from online courses, technical forums, blog posts, and wikipedia. Recognizing this, you are encouraged to make *careful* use of these resources. They can be valuable references but their quality and notation can vary. Ultimately you will be tasked with developing solutions to novel problems in your careers so it is critical that an understanding of the theory of a solution is developed rather than simply stringing together code snippets from stackoverflow. There are no restrictions on the resources you makes use of, but resources must be clearly documented, including a way for me to access a resource, a description of what was obtained from each resource, and how it was used.

If you use code snippets which are not original work, the snippets should be accompanied by referenced comments identifying the source of the snippet in a way I can access (url) and a 1-2 sentence explanation of what that snippet does. For example:

```
# uniformly sample initial position on earth. I knew that sampling randomly from
# latitude and longitude would over-sample the poles and under-sample the
# tropics. I figured that the earth is close enough to spherical that if I
# sampled on a sphere and normalized it to the earth's size that that would be
# close enough, but couldn't think of how to do this sampling, I found this
# solution on stackexchange and translated it from R code to matlab.
# https://stats.stackexchange.com/q/7988
n_samples = 1000
z_sample = 2 * np.random.rand(n_samples, 1) - 1
theta_sample = 2 * np.pi * np.random.rand(n_samples, 1) - np.pi
x_sample = np.sin(theta_sample) * np.sqrt(1 - z_sample**2.0)
y_sample = np.cos(theta_sample) * np.sqrt(1 - z_sample**2.0)
R_earth = 6378137.0
x = x_sample * R_earth
y = y_sample * R_earth
z = z_sample * R_earth
```

Note that you are still responsible for your solution being correct and working. For example – before I put this example in I did some basic checks (plotting the output and checking the statistics against a uniform lat-lon sample to make sure they were different).

## Course Structure and Sequence

The course will largely be structured around a series of seminal papers in robotic planning and decision-making. There will occasionally be a lecture to introduce some material, but in general you will read the assigned paper and then we will discuss it and work through key algorithms as a class. There will roughly be three periods of this class

### Preliminaries

In the first part of the course we will cover required mathematical background. This will be where most traditional lectures are, we will:

- cover basic graph theory
- develop a notion of “optimality”
- distinguish between paths and trajectories

### Planning

This will occupy the majority of the course. We will aim to work our way through a major planning algorithm every two weeks. For each algorithm you will be assigned to read the paper developing it. We will then discuss the paper as a class and work through the algorithm by hand to develop an understanding of the algorithm’s mechanics. Meanwhile you will implement a computer code for the algorithm and evaluate it on test cases I will provide. We will cover:

- finding paths on graphs
- improving planning efficiency using heuristics
- replanning efficiently
- using primitives
- planning with dynamical constraints
- connections between planning and optimal control
- planning in high dimensional spaces
- planning for uncertain systems

### Decision-Making

Finally, we will explore connections between planning and decision-making. These occur both in the problem structure, and in ways that we can use both planning and decision-making in the same problem. We will cover

- probabilistic decision-making
- connections between planning under uncertainty and decision-making

- interfacing planning and decision-making systems

## Assignments and Evaluation

Progress in this course will be evaluated through a sequence of exercises involving implementing solutions to planning problems. You will produce mini reports for each exercise and provide a self-assessment. Your mini reports should cover:

- the model for your system, if appropriate
- design choices you made during implementation of your planner
- analysis of possible failure modes
- analysis of performance of the planner

Reports should cover no more than three total pages including figures and equations but excluding references. Any resources used in designing or implementing solutions to the activity problems should be documented.

You should also turn in commented source code. Comments should not describe what a line of code does, but rather why you are doing something. For example, this is an example of a good comment:

```
# I make a forward euler approximation in the state dynamics. This allows me to
# use the continuous-time dynamics matrix I've derived to propagate the discrete
# time dynamics. This will break down if my time step sizes get too large or if
# the dynamics are very fast.
A_step = np.eye(2) + A * dt
X_ = A_step @ X_ + B @ u * dt
```

While this is not an adequate comment:

```
# propagate the state vector forward
A_step = np.eye(2) + A * dt
X_ = A_step @ X_ + B @ u * dt
```

You will perform a self-assessment on each exercise. One objective is for you to not only know and understand the material, but to have an understanding of how well you know and understand it. You will complete the self assessment so that you can later compare it to feedback I provide, helping you to calibrate yourself.

## Project

You will also complete a self-identified project requiring application of the concepts and practice of planning and decision-making. You are encouraged to draw the project content from your research, personal, or professional interests. You should discuss a project topic with me and we will agree on an appropriate scope and deliverable. Projects may work with real or simulated systems but require:

At the conclusion of the semester you will turn in a final project report no more than 10 total pages

(excluding references). Source code should be provided and follow the same documentation requirements as the exercises.

You will be provided suggested milestones. I very strongly recommend you satisfy them. This is a major undertaking and will take you several weeks to complete satisfactorily

## Grade Expectations

Rubrics for evaluation of course assignments will be constructed so that a “C” level grade indicates a satisfactory solution to the assigned problem but without demonstrating an understanding of the theoretical background for the problem and its solution. A “B” level grade will indicate both solution of the assigned problem and an understanding of its theoretical properties. “A” level grades will indicate that in addition to “C” and “B” level mastery that you understand the limitations of the solution developed.

## Grade Assignment

This is a graduate level seminar class. My concern is that you engage seriously with this material, find where you can take it into your own interests and research, and learn to assess your own competence and understanding. Evaluation of your work is important, but grade assignment contributes relatively little to this objective. To this end we will experiment with “ungrading” this course. I will provide nominal weighting and rubrics and ask you to assign yourself scores. I reserve the right to intervene and assign scores based on the rubric.

### Nominal Weights

Deliverable	Weight
Exercises	0.75
Project	0.25

## Preliminary Schedule

Day	Exercise	Topic	Relevant Text
M, 2022-08-22		Introduction and python exercises	
W, 2022-08-24		Lecture – Paths vs Trajectories, Cost Functions	

Day	Exercise	Topic	Relevant Text
M, 2022-08-29	python	Lecture – Graphs, I	
W, 2022-08-31		Lecture – Graphs, II	
M, 2022-09-05		Labor Day, No Class	
W, 2022-09-07	1	Dijkstra's Algorithm	
M, 2022-09-12		Dijkstra's Algorithm	
W, 2022-09-14		Dijkstra's Algorithm	
M, 2022-09-19	2	Lecture – Heuristics	
W, 2022-09-21		A*	
M, 2022-09-26		A*	
W, 2022-09-28		A*	
M, 2022-10-03	3	D*	
W, 2022-10-05		D*	
M, 2022-10-10	4	Dubin's Car	
W, 2022-10-12		Dubin's Car	

Day	Exercise	Topic	Relevant Text
M, 2022-10-17		Lecture – Planning for Dynamic Systems	
W, 2022-10-19	5	Optimal Control	
M, 2022-10-24		Optimal Control	
W, 2022-10-26	6	Lecture – High Dimensional Planning, last class before drop day	
M, 2022-10-31		RRT	
W, 2022-11-02		RRT	
M, 2022-11-07		RRT	
W, 2022-11-09	7	Lecture – Decision-Making	
M, 2022-11-14		Probabilistic Graphs	
W, 2022-11-16		Probabilistic Graphs	
M, 2022-11-21	8	Probabilistic RRT	
W, 2022-11-23		Probabilistic RRT	
M, 2022-11-28		Decision-Making over Plans	
W, 2022-11-30		Decision-Making over Plans	



# Course Policies

## Technology Requirements

Homework, activities, and projects will require computer software programming. You will be provided with python environments and shells for the exercises. You can [install python by itself](#) or as part of [the anaconda environment](#). Unless you have been regularly using python in the past I recommend using anaconda.

You will have to install some additional libraries to complete the activities. At a minimum you will need:

- numpy
- scipy
- matplotlib

These all come with anaconda, so if you go that route you will be set.

If you do not have access to a laptop, you can borrow one from [the library at this link](#).

Activity and project reports should be submitted as zip files containing a PDF report and a “software” folder through the blackboard system. You may submit your code via links to a git repo that I can access if you prefer.

## Course Attendance Policy

There will be no formal attendance taken, but you are expected to be engaged in the class discussions.

## Late Work Policy

Assignments may be accepted at my discretion provided that you contact me more than 24 hours in advance and receive an extension. Absent an emergency or prior extension, work that is simply not turned in on time will not be accepted after the deadline. If you have an emergency get in touch as soon as practical and we'll work something out.

## Accommodations Policy

The University is committed to providing reasonable accommodations and auxiliary services to students, staff, faculty, job applicants, applicants for admissions, and other beneficiaries of University programs, services and activities with documented disabilities in order to provide them with equal opportunities to participate in programs, services, and activities in compliance with sections 503 and 504 of the Rehabilitation Act of 1973, as amended, and the Americans with Disabilities Act (ADA) of 1990 and the Americans with Disabilities Act Amendments Act (ADAAA) of 2008. Reasonable accommodations will be made unless it is determined that doing so would cause undue hardship on the University. Students requesting an accommodation based on a disability must register with the UTEP Center for Accommodations and Support Services (CASS). Contact the Center for Accommodations and

Support Services at 915-747-5148, or email them at [cass@utep.edu](mailto:cass@utep.edu), or apply for accommodations online via the CASS portal.

## COVID 19 Precautions

You are expected to adhere to university guidance on COVID 19 precautions available at: <https://www.utep.edu/resuming-campus-operations/> Guidance and policy with respect to COVID may change throughout the semester.

If you have tested positive for COVID 19 or have reason to suspect you may have COVID 19 (e.g. because of symptoms or close contact with an individual who has COVID 19) you are expected to stay home as directed by [CDC guidelines](#). Contact me and we will arrange appropriate accommodations.

Students who are considered high risk according to CDC guidelines and/or those who live with individuals who are considered high risk may contact Center for Accommodations and Support Services (CASS) to discuss temporary accommodations for on-campus courses and activities.

## Academic Integrity

Academic dishonesty is prohibited and is considered a violation of the UTEP Handbook of Operating Procedures. It includes, but is not limited to, cheating, plagiarism, and collusion. Cheating may involve copying from or providing information to another student, possessing unauthorized materials during a test, or falsifying research data on laboratory reports. Plagiarism occurs when someone intentionally or knowingly represents the words or ideas of another as ones' own. Collusion involves collaborating with another person to commit any academically dishonest act. Any act of academic dishonesty attempted by a UTEP student is unacceptable and will not be tolerated. All suspected violations of academic integrity at The University of Texas at El Paso must be reported to the Office of Student Conduct and Conflict Resolution (OSCCR) for possible disciplinary action. To learn more, please visit HOOP: Student Conduct and Discipline.