# Chapter 5.3 - Linear Programming

11 MIN · YOUTUBE · HTTPS://WWW.YOUTUBE.COM/WATCH?V=MBBMUNC2NMY
https://www.youtube.com/watch?v=mbbMUNC2NMY

SUMMARY

*Linear programming is a method used for solving constrained optimization problems where the objective function and constraints are linear. This approach is essential in various fields such as engineering and science, as it allows for the systematic handling of constraints that reflect real-world limitations.*

- *Linear programming involves minimizing a linear function subject to linear constraints.*
- *Constraints can be equalities or inequalities, ensuring that control variables remain within feasible limits.*
- *Slack variables are introduced to convert inequalities into equalities, facilitating the optimization process.*
- *The fundamental theorem of linear programming states that if a feasible solution exists, a basic feasible solution can also be found, often with many variables set to zero for simplicity.*
- *The optimization process can be implemented using programming languages like Python, which have built-in libraries for linear programming.*
- *The solution space is defined by the constraints, and the optimal solution is typically found at the vertices of this feasible region.*
- *The method is widely applicable and can be easily implemented in various coding environments, making it accessible for practical applications.*

we turn our attention now to thinking about optimization where we would actually have constraints and one of the most standard ways to think about solving constraint optimization is through what's called linear programming of course this is a a subset of all these constraint optimizations as the name implies linear programs are going to be something where we have linear constraints as well as some kind of function that's a linear optimization of the control variable so we're going

to set this up and you're going to see that this is sort of something that we can syst systematically go about doing so for constraint optimizations unlike what we did previously with unconstrained we actually have constraints on the control varibles so we start off with something we want to minimize but now it's subject to and this Subud line becomes very important because constrainted optimization is is commonly used across engineering and and Science and so we want to be able to

start enforcing things like there's some function of X that's equal to zero or an inequality G of X greater than zero something like this that we can impose on the system that relates to the reality for instance if you're trying to solve for the optimal concentrations of gases to for for some mixture all the gas gases have to have positive densities right we cannot come up with some kind of gas or version of this where we have some

unphysical characteristic of the system so the constraints help us find models where we enforce that mass is positive concentrations are positive things like this that are really reasonable to enforce okay so let's come up with a model for this the control optimization this is a linear model with constraints so linear programming is going to handle this kind of architecture so notice what we have here we first in the first line we have what we're going to try to

minimize which is a linear combination of the control variables with some constants that are given C1 through C of n so that is the function we're going to try to minimize and so we have to figure out how do we choose X1 X2 all the way through X of n to minimize that so what value of those control variables minimize that but now notice how much more complex this is than unconstraint optimization because the first one if I didn't have

constraints I could just do gradient descent but now I have these constraints which is I may have M constraints listed here these are exact constraints so in other words they're not inequalities they are exactly held but also some kind of inequalities that I might have that for instance X1 through X of n are all greater than zero so they have to be positive so this is a very generic form of what we want to start thinking about

how to optimize this so youve trying to find the solution subject to all of this being true okay so the way we typically would write this is in a more linear algebra way which is the following right so minimize cpose X so c c is the those constants C1 through C of n acting on X this is this is the function that I'm trying to minimize shev to some ax equal to B and some constraint where X all the

values of X are of control VAR are positive greater than or equal to zero okay so this is the generic linear model that we want to work with and so what we can start to think about doing is in in fact in many constrainted optimization problems it's actually more formulated like this where we actually have constraints that are inequalities right here so for instance these M equations all are equal to or less than some constant B and so what

we're going to do in this context is we're going to solve this by introducing what we called slack variables so if this is less than or equal to B1 then what I'm going to do for each one of these M constraints I'm going to introduce a slack variable y1 through y of M and I'm going to make this from an inequality to an equality okay so that's what the slack variables going to introduce to us the slack variables y1

through y of M all have to be positive for me to go through this inequality these y's have to be positive but now what I have is extra variables available to me now remember these constraints if I have an M constraints on on N variables I sort of in some sense have an underdetermined type system what that means is that I have a lot of flexibility a lot of solutions I can work with and what I'm trying to do is

find the ones that are best for this problem out of an in in Infinity of possibilities of feasible Solutions so this is what we're going to to solve with the slack variables and in facts it leads us to this statement here about it's the fundamental theorem of linear programming so this is important for us so if we set up our linear program in this way which we've just written down here the fundamental theorem of linear program says a given a linear program in the in this form then there exists a feasible solution if if there exists a feasible solution there also exists a basic feasible solution so what are the difference a feasible solution is something that actually just satis y all this a basic feasible solution means that I can actually because this is an underdetermined system I try to put in as many zeros as possible so if I can find a feasible solution I can transform it to making as many of the variables

zero as possible that's the basic feasible solution and if there's an optimal feasible solution then there's an optimal basic feasible solution in other words with as many zeros is possible so this is very nice for us because it's going to structure how we're going to work with those Slack variables cuz we're going to use them to not only find feasible Solutions but then find those feasible Solutions with the most number of zeros sort of in some

sense they're most interpretable as possible so let's start off with a given example here and then we're going to actually use Python's linear programming capabilities to solve it so here's what we're trying to do we're going to minimize this function here - 2x1 - X2 here are the constraints we have we have three constraints okay and subject to the these two being bigger than zero so what we're going to do to this is we're going to add slack

variables since I have these three constraints I add slack veral X3 X4 and X5 these are just things I introduced but I noticed what I've done here is now I've turned this to an equality where I've made the X3 X4 and five all have to be positive so I've essentially made the problem go from a two-dimensional variables to five where I've introduced these artificial variables X3 X4 and X5 but the power now of that is that I can

look at these are exact constraints by introducing the slack variables in fact what I can do now is look at when is X3 equal Z when is X4 equal to Z when is X5 equal to zero and these set up lines in the parameter space so let me show you the example here for this specific example if I look at X1 and X2 these dotted lines here that you see those are setting the slack variables equal to

zero and what does it enforce about X1 and X2 so when X2 equals 0 you get this slope line here here is X5 equal to Z here's X4 equal to Z and what that does is set the boundaries of our feasibility region which is this gray box so in this gray box by me using the slack variables I constrain everything into this gray box and now I can just evaluate the function across this gr box and you can see that now this is the

visibility region and if I go to this corner right here this is actually the solution where X3 = 0 X4 = 0 X5 equal 0 in other words in this region X3 X4 and 5 are not zero so that's a they're feasible Solutions but they're not basic feasible Solutions when I walk to the edge of this box right here what I have is a basic feasible I have a basic optimal feasible solution in fact the value there is

1.5 1.5 okay so how do we set this up in a linear program so this is the basic architecture here we have ax less than b or we set it up with slack variables in this form here right so this is what we just did we introduced slack variables to get us into this form and so if we go back to the problem we've been looking at here it is in this form here what we're going to do is set up a set of matrices to Define this and here's how it is and so we're going to think about extending this into slack variables three slack variables would go here so when we look at these constraints which are given by here I I could write them as three constraints but if I go up to the five variables right the slack variables then what I have here is equalities and that's what we're going to do here is write this as an equality

for X1 X2 X3 X4 X5 okay so that's what I'm setting up here is the Matrix a which is a x x is now just not X1 X2 it's all the way to X5 here's B this is C so all you've got to do is in this setup here is give me these matrices a b and c once you have them this is as easy as as it is to use linear program for from s pi optimize

you can bring in linear program you set up these matrices and then you just say linear program there you go C A and B and then when you run that there's your answer okay so a linear program will go through and do exactly what we've talked about here it will actually so in this way we've set it up we've introduced the slack variables and introducing the slack variables we can just put this into this form this a and C's it will actually go

through look through the feasibility sets find the optimal feasible solution and it will always try to find the basic optimal feasible solution notice this it's not spitting back values of x3 X4 X5 because it's assuming I can get those to be zero at the end of this okay so in other words I want the basic solution so that is how you do a linear program it's a it's a important method if if your problem Falls within

the constraint structure that I've written down there of the of the linear program then it's a very easy thing to implement because almost any code package actually already has it built in so it's very easy to use so that is the introduction to linear programming or if you want to think about this as the first introduction we have to constrainted optimization problems