[SQUEAKING]

[RUSTLING]

[CLICKING]

PETER SHOR: So last time we defined canonical forms of a linear program and their maximize c transpose x where these are vectors with such that-- sorry, subject to Ax is less than or equal to b, and x is greater than or equal to 0. And the dual for the canonical form.

Dual, we showed was minimum b transpose y. So we've taken the right-hand sides of the equation and turned them into the objective function. A transpose y is greater than or equal to c. So we've taken the objective function of the primal and turned it into the right-hand side of the equations of the dual. And y greater than or equal to 0.

And last time, we've showed weak duality for any feasible x in the primal and any feasible y in the dual solution. Now, feasible just means they satisfy the constraints. c transpose x is less than or equal to b transpose y. So this is true for when the optimal value is finite.

So a linear program can have finite optimum. It can be infeasible. What's an infeasible program? x1 greater than or equal to 5 and x1 less than or equal to 3. So this is a basically trivial example of infeasible program. There are no solutions. Most infeasible programs, it's much harder to-- you cannot tell that it's infeasible just by looking at it.

And it can be unbounded. And let's give you an example of that. Maximize x greater than or equal to 3. You can see that you can find a feasible x such that it has any value for the objective function. And so these are-- actually, it's pretty clear that these are the only three possibilities.

OK. Now I'm going to tell you, well, something that follows immediately from weak duality. If the primal is unbounded, the dual is infeasible. And that's because if there were some feasible solution to the dual, then you could find a solution to the primal which is larger than that.

And this is the contradiction of weak duality. And similarly, if dual is unbounded, the primal is infeasible. And there's one more possibility. Possible dual, primal are both infeasible. So this doesn't happen very often, but it can happen and you probably should be aware of it. We're not going to discuss it any further.

Sometimes we give a homework problem asking you to find an example where this is the case, and it's not that hard a homework problem. So if you want to figure out how this can happen, you can go home and figure it out. Or you can probably look it up on Wikipedia and find it that way. OK. So first thing I want to do today is talk about what the primal and dual look like for things that are not in canonical form.

And I'll first give you the treatment and the lecture notes, and then I will give you my own treatment, which is slight-- well, it's not really different but it's slightly different. And I think it makes it easier. So let's look at the proof of duality. Maybe I should do this by example, because I like doing things by example, because it's easier than-- sometimes it's easier than doing them abstractly.

So let's say max 3x1 plus 5x2 minus 2x3. x1 plus x2 less than or equal to 8. x1 minus x3 is greater than or equal to 1. And x2 plus-- OK, I did this wrong. I want to leave a gap here so you can line up the variables. Greater than or equal to 1. And x2 plus x3 equals 4. Recall proof of duality.

The dual has a variable for each constraint in the primal, and a constraint for each variable in the primal dual. Each primal variable. Oh, I'm sorry. I left out the non-negativity conditions. X1 and x3 are greater than or equal to 0, and x2 is unconstrained. So there's a dual variable for each primal constraint.

Good. And now what we did was we took the inequalities, multiplied them by the new variables and the dual, and added them up. So we have y1 times x1 plus x2 is less than or equal to 8. y2 times x1 minus x3 is greater than or equal to 1. y3 times x2 plus x3 equals 4.

OK. And when we add them up, we want these variables to-- or we want these inequalities to all point the same way. Because if we add x1 plus x2 less than or equal to 8 and x1 minus x3 greater than or equal to 1, it doesn't tell us anything. So that means that here we have to have y1 greater or equal to 0. Here we want to flip the sign of this inequality, so we need y2 less than or equal to 0.

And here, well, it doesn't matter whether y3 is bigger than or less than or equal to 0. Because when we multiply it, there'll be an equality. And when we add them all up, it'll still be the right inequality. So here, we have y3 unconstrained. OK. So going from a less than or equal to sign, if the primal is a maximization, gives you a positive variable. A greater or equal to sign, if the primal is maximization, gives you a negative variable, and an equal sign gives you an unconstrained variable.

OK. Now. Remember, we wanted to maximize 3x1 plus 5x2 minus 2x3. So the first column gives you y1 plus y2 is greater than or equal to 3. The second column gives you y1 plus y3 is greater than or equal to 5. Is that right? And the third column gives you minus y2 plus y3 is-- minus y2 plus y3 is greater than root 2 minus 2.

So these are the constraints of the dual. And similarly, when you're going from the dual to the primal, everything works the same way, except less than or equal to-- greater than or equal to signs turn into positive variables, less than or equal to signs turn into negative variables, and again, equal signs turn into unconstrained variables. So I don't like doing it this way. Yeah, you have a question?

AUDIENCE: Oh, shouldn't y1 plus y3 be equal then [INAUDIBLE]?

PETER SHOR: Yes. Yes, it should. y1 plus y3 equals 5. Thank you. OK. So what was I going to say? I was going to say, the reason I don't like this is it gives you negative variables, and negative variables are hard to think about. And when you're thinking about them, you're liable to make mistakes. So what I'm going to do is I'm going to-- yeah. Another way to take the dual.

First, make all equations the correct less than or equal to or greater than or equal to, the correct sign. So here what we want to do is we want to take the second equation and turn it into a less than or equal sign, because if we have a maximization and canonical form, you want less than or equal to signs.

So that turns into 3x1 plus 5x2 minus 2x3, subject to x1 plus x2 less than or equal to 8 minus x1 plus x3 is less than minus 1 and x2 plus x3 equals 4. And I forgot to write down x1, x2 greater than or equal to 0 No, x1, x3 greater than or equal to 0. x2 unconstrained.

So now we can just write this off. We take the transpose of the A matrix, which is y1 minus y2, y1 plus y3, and y2 plus y3. x2 was unconstrained, so this is an equality. And x2 and x3 and x1 both had greater than or equal to signs, so these are less than or equal to signs. The minimum, because the original, the primal was maximization.

The dual is minimization. And we just read off the right-hand sides of the equations, the inequalities, and we get 8 y1 minus y2 plus 4y3. And we call this w, not that it really matters, subject to these things. And we take the original objective functions and use it for the right-hand side. So 3, 5, and minus 2.

And finally, a third equation was an inequality, so y1, y2 greater or equal to 0, and y3 unconstrained. And we don't have this hard to think about variable y2 less than or equal to 0. So this is the way I like to take duals. OK. So the next thing I want to do is talk about complementary slackness. Any questions about the materials so far? OK.

Complementary slackness, we are going to-- well, why don't I write the theorem down and then we'll prove it? So again, we're going to have two linear programs, and we're only going to prove it for the-- we're going to prove it when the linear programs are in canonical form. But in general, it's true for any linear program.

Suppose x is a feasible solution in the primal and y is a feasible solution in the dual. Then x is optimal in primal and y is optimal in dual if for all i, either yi equals 0 or A sub xi equals b sub i. So this is either saying the i-th variable in the dual is 0 or the i-th equation in the primal is satisfied with equality.

And for all j, either x sub j equals 0 or A transpose yj equals b sub j. So this is saying either the j-th variable is 0 or the j-th equation in the dual-- OK, this is not b. This is c. Either the j-th equation in the j-th variable is 0, or the j-th equation in the dual is satisfied with equality.

OK. So we'll prove it. And we'll only prove it for-- or rather, we will prove it for linear programs in the canonical form, but because it's true for linear programs in the canonical form, it's automatically true for all linear programs. There's no reason you have to prove it for an arbitrary linear program, and it's just extra notation. OK. So we should write down the programs in the canonical form.

And maximum z equals c transpose x, Ax less than or equal to b, x greater than or equal to 0. So this is the primal. And the dual is the minimum w equals b transpose y, A transpose suppose y equals c is less than or equal to c, and y greater than or equal to 0.

OK. So what we have is, then proof. Well, we have A transpose y is less than or equal to c. So let's start with c transpose x, and now let's plug this equation into here. c transpose x is less than or equal to-- well, c transpose is y transpose A. So this is y transpose Ax.

Now, y transpose Ax is equal to y transpose Ax because matrix multiplication is associative. And we have Ax less than or equal to b. So this says it's less than or equal to y transpose b. So this is just weak duality. Suppose both of these, both less than or equal to signs hold with equality. Well, then we know that for this to be optimal, these signs need to be held with equality.

And if this holds with equality, well, then let's rewrite this equation y transpose A minus c, x equals 0. Well, this should be c transpose. And let's rewrite this equation. That's just y transpose Ax minus b equals 0. OK. So if we have two dot products of vectors are 0 and each of these vectors contains only positive entries, then every piece of the dot product, one of these two equals 0.

So y transpose i equals 0 or Ax sub i equals b sub i. And here, it's just the same thing. Either xj equals 0 or y transpose Aj y transpose A sub j equals c transpose j. So this is one way of the inequality. Now suppose the other way holds. That means either xj equals 0 or y transpose Aj equals cj.

Well, that means that this product is 0, which means that this is an equality. And similarly, on this side, if either this equals 0 or this equals 0, then this dot product is zero, and that means that this is an equality. So z transpose x equals y transpose b, which means that c transpose x and y transpose b are optimal solutions. OK. Now I want to go back here and tell you something neat you can do with complementary slackness.

OK. So x1 equals 4, x3 equals 4 is a feasible solution. Right? Because 4 plus 4 is less than or equal to 8 minus 4 plus 4 is what? x1 plus-- this is an x2, not an x3. 4 plus 4 is less than 8 minus 4. Oh, I forgot. x3 equals 0. Minus 4 plus 0 is less than or equal to minus 1, and 4 plus 0 equals 4. So this is a feasible solution.

And in fact, it's the optimum solution. And if you know it's the optimum solution to the primal, you can use complementary slackness to find the optimum solution to the dual. So here, we have two variables which are not equal to 0, which means we have two equations which are satisfied with equality.

So that means that we must have-- so the two variables which are not equal to 0 are the first and the second, which means the first and the second equation are satisfied with equality. So y1 equals y2. OK. I hope I'm doing this right. No. I'm not doing it right, because this equation is y1 minus y2 is less equals-- equals 3. So y1 minus y2 equals 3, and y1 plus y3 equals 5.

Now up here there's an equation which is not satisfied with equality in the optimal solution. That's the second equation, which means y2 equals 0. And now you've written down these three equations, and there are three unknowns, so you can solve it. And so y1 equals 3 and y2 equals 5.

And now you can plug this in and see whether this equation for the primal or this feasible solution for the primal and this solution for the dual give you the same objective function. So let's do that. We have x1 plus x2 is 3 times 4 plus 5 times 4. That's 32. So objective function, 32. And down here, we have 24.

OK. y1 equals 3. I obviously did my equations wrong. y1 equals 3. y2 equals 0, so y1 equals 3. And we have 3 plus y3 equals 5. So y3 equals 2. Apologies for my arithmetic. So that means that we have 24 plus 8 is 32. Objective function equals 32. So this is an optimal solution for the primal and this is an optimal solution for the dual.

OK. So the last thing I want to do is give a-- I don't know if I want to call it a proof or not, because maybe I give a justification for strong duality. If you look online, there are roughly two kinds of proofs. There are three kinds of proofs of strong duality. The first, you define the simplex algorithm, which is a way of getting the optimal solution for linear programs.

And then when the simplex algorithm finishes, it gives you what's called a tableau, from which you can read out the solution to the primal, you can read off the optimal solution to the dual, and then you can stare at it and use the-- well, you can-- it tells you that the optimal solution to the primal is equal to the optimal solution to the dual. So that's one way of proving strong duality, and it's absolutely rigorous.

Another way of proving strong duality is borrowing this lemma from analysis called Farkas' Lemma and applying it. The problem with this is that Farkas' lemma is basically strong duality in different notation. So you haven't really-- I mean, I guess it's a proof if you believe Farkas' lemma, but it doesn't seem to give you any more real insight than my just telling you that strong duality holds, because that's what it's doing.

And the third uses mechanics in n dimensions. We need a few basic theorems from mechanics, and then we can prove strong duality. So, I mean, we have to accept these theorems from mechanics. I don't know if there's any way going through mathematics and proving that these theorems from mechanics really hold, but I mean, if you accept the theorems from mechanics, then you absolutely have a proof of strong duality and showing--

I mean, I find this really pretty amazing, showing that mechanics is consistent in high dimensions. It's the same thing, in some sense, as proving strong duality of linear programming, which is a connection I never would have expected. So we're going to prove give you the justification from mechanics. And let's see.

So a linear program set of hyperplanes. The constraints. The constraints say you're on one side of the hyperplane. And the objective function, you want to find feasible point furthest in direction of objective function.

So what do we do? Well, we pretend each of these hyperplanes is a solid surface. OK. This is a bad drawing. And what we're going to do is we're going to drop a ball, and gravity is going to take it in the direction of the increasing or decreasing objective function, and it's going to hit this hyperplane. And now it's going to roll in this direction and it's going to hit this point.

And now we're going to assume-- well, actually, maybe we should make another plane here. It's going to roll down to here, and this is the lowest point on the proper side of all the hyperplanes so it's just going to sit there. And mechanics tells us that the force of gravity on it is balanced by the force being pushed on this point perpendicular to each hyperplane.

So the sum of the forces exerted on this ball by the hyperplanes is equal to the force of gravity. And this is a theorem from mechanics we need. OK. So now let's write down the linear program that solves this problem. Minimize c transpose x subject to Ax is greater than or equal to b.

And this is just saying the point is on the correct side of every hyperplane. And this is just the force of gravity. You want to make the point as low as possible. And, well, x is an arbitrary point, so x is unconstrained. So now we can ask, what's the dual of this problem?

So this is the primal. The dual, well, it's going to be max v transpose y. And now all of these x's are unconstrained. So that means all of these equations must be equalities. So we have a transpose y equals c. And here, we have a whole bunch of inequalities which means we get y greater or equal zero.

OK. So this is the linear program and its dual, that we will show how the property of strong duality. So a hyperplane. The equation was-- I left out some letters there. The equation was Ai1 x1 plus Ai2 x2 plus dot, dot, dot plus Ai n x sub n is greater than or equal to b sub i.

So what is the upward force on this hyperplane? Well, if you have a hyperplane and you have a ball resting on the hyperplane, the force on the ball is perpendicular to the hyperplane. So it's in direction. I want to say Ai1 Ai2 through a sub i n. It has to be positive, so it's some constant c. So force is c Ai1 through A sub i n with c greater than 0.

OK. c is a bad number for this, bad variable for this. Let's see. Let's call it f. I should probably call this f sub i. OK. OK. OK. We have there must be some forces.

And I'm going to call them y sub i instead of f sub i, just because y sub i is the right name for the variable because it's a dual variable, such that summation-- OK. Such that if v sub i is equal to this direction, Ai1, Ai2 through A sub i n, then summation v sub i. No. Yeah, that's right. Then summation y sub i, v sub i is equal to c.

Actually, maybe we should make this c transpose. And because these forces all act away from the hyperplane, we need y sub i greater or equal to 0. OK. If the ball touches the i-th hyperplane, then A sub i. If the ball touches the i-th hyperplane, I want to say Ax.

Ax. The i-th term of Ax is equal to b sub i, because if the ball touches the i-th hyperplane, then the constraint is satisfied with equality. And this is true. The ball does not touch the i-th hyperplane. Then what's going on? If the ball does not touch the i-th hyperplane, then the hyperplane does not exert any force on the ball.

Then y sub i equals 0. And these two things are enough to give you complementary. Are enough to-- I mean, these two things, first, look a lot like complementary slackness, but they're enough to give you strong duality.

OK. OK, we have y sub i. No. y transpose Ax minus b is equal to yi, I guess summation y sub i, Ax minus bi. And I want to claim either yi equals 0 or Ax sub i equals b sub i. So this equals 0. Now this goes y transpose Ax minus b equals 0. And let's look back at the linear program.

And we have A transpose y equals c. So y transpose A is c transpose. So this says-- oh, this should be y transpose b. So when I took this, I forgot to move the y transpose into the parentheses. So this is c transpose x, and so we have c transpose x minus y transpose b equals 0 and c transpose x equals y transpose b.

OK. So that actually is the end of the linear programming duality lecture notes. And I suspected I might have a little time left, so I am going to tell you about Konig's theorem, which is an application of linear programming duality. And I did not prepare this as well, so hopefully I won't make any mistakes.

Konig's theorem. Suppose you have a bipartite graph. OK. So let's draw an example. OK. Matching is a set of edges that do not-- I want to say that do not meet at any vertices.

So this is one edge of a matching. This might be another edge of the matching. This might be another edge. And here's another edge. So that's four edges in this matching. OK. Now I want to claim-- now I want to define a vertex cover. Vertex cover is a set of vertices that intersect all the edges.

That meet all the edges. So let's see. What would a vertex cover in this graph look like? Well, you might use this vertex. You might use this vertex. You might use this vertex. And you might use this vertex. Does this cover all the edges? No, actually, it doesn't. So let's not use that vertex. You could use this vertex. And that actually does cover all the edges because-- yeah.

OK. Theorem. The size of the maximum matching is equal to the size of the minimum vertex cover. OK. So what we have here is we have four in each of them. So this is an example where equality holds. And the Konig's theorem says equality holds in all graphs.

And there's one which one direction is easy. Size of any vertex cover has to be bigger than or equal to the size of any matching. And why is this? Well, I mean, a vertex cover has to cover each of the edges in the matching, which means it has to be one of the endpoints of each of the edges of the-- has to contain one of the endpoints in each of the edges of the matching, which means that x has to be the size of the maximum matching, at least.

And Konig's theorem says that they're equal. OK. So how do we prove equality? Well, one way to do it is write down a linear program for the maximum matching. Write down a linear program for vertex covers. Show that they're duals, and then show that there is an integer solution to each of them.

So I do not have time to do all of that, but we can at least write down the theorem for the maximum matching. How are we going to write down a linear program for the maximum matching? What should variables be? Edges, maybe? OK.

So we want to maximize the sum of x sub e. OK. And now we want to say that at most one of these edges-- if we have a vertex here, we want to say at one of these edges is present in the matching. And we're not going to be able to do that because these are not integer variables. They're real numbers. But let's write down zero.

That's an xe lesson one and summation v is contained in e. So for every vertex, the edges containing that vertex x sub e is less than or equal to 1. So this is something like there is at most one edge containing every vertex. Which means that we do not have two vertices, two edges intersecting in vertex and are matching. Assuming that all the variables are 0 or 1, and that's exactly the criterion for the maximum matching.

OK. Let's find the dual. Well, the dual-- I mean, we have an equation for every vertex and some extra equations and a maximization. So the dual is going to be a minimization. It's going to have an equation of variable for every edge, and it's going to have some extra variables. Dual.

OK. So one constraint per edge. And here, we have one per vertex. Here we have extra per constraint per edge. And I guess we can also have non-negative constraints. So let's see. The dual is going to have-- I should write down what these equations-- what these variables are. Dual. Let's call this y sub v and z sub e, the equations.

Now dual equation for each. Our inequality, I should say. For each edge. And what are these inequalities going to be? Well, y sub v appears in the equation for x sub e, and z sub e appears in the equation for x sub e. So that means that y sub v plus z sub e.

Well, let's see. Minimum summation y sub v. y sub v plus z sub e is less than or equal to 1. Oh, wait. This is greater than or equal to. I'm sorry. And yv and z sub e. So what does this really mean? Well, what you want is you want a vertex cover, which means you want a set of vertices that are adjacent to every edge.

OK, so actually, I did that wrong. Minimize y sub v plus z sub e. Because the right-hand sides are 1 for y sub v and 1 for z sub e. OK. OK. I seem to have gotten myself lost here, but I can go back and talk more about the program for the primal program. And why the primal program is-- yeah. So that's the primal program up here.

So the primal program. A solution is a weight on each edge. So some of the weights, we, v in e, is less than or equal to 1. So suppose we have fractional solutions. That means that there's some edge where you might have two-- some vertices where you might have two edges, which have 0.5 going out of it.

Now, we might have another edge going out of 0.5 out of this one, because this plus this is less than or equal to 1, and we want to maximize it. So we would have another one going out. And this might be 0.25. Let's put one up here. Maybe another 0.5 and another 0.5.

OK. In a fractional solution, you can always find a cycle of fractional edges. Right? Because, well, I mean, if you cannot find a-- if you find a cycle of fractional edges, what you can do is you can increase the even ones and decrease the odd ones and set one of them equal to 0 and always get a better solution.

And first, all of these cycles are even because this is a bipartite graph. So that means we can increase the odd ones and decrease the even ones and we get 1, 0, 1, 0, 1, 0. And it has the same objective function and fewer fractional edges. In doing this, we can always get rid of the fractional edges. Now, there's another possibility, which is you have 0.5, 0.5, 0.5, 0.5. And what you can do is you can now increase this one to 1, decrease this to 0, increase this one to 1, decrease this to 0.

Because if you can find a path, that would mean that this vertex does not have any fractional edges next to it aside from this one, and this vertex does not have any fractional edges next to it besides this one. And that means that you can increase the odd edges and decrease the even edges. Either you end up with the same objective function or you increase the objective function.

But we're assuming this was optimal, so you always end up with the same objective function. So you can always decrease the number of fractional edges and keep the same objective function, which means that eventually you find a solution where there are no fractional edges, and that is your maximum matching that you wanted to find.

And if you take the dual of this and stare at the dual, you will see that it gives you-- and I screwed that up when I was trying to do it, but it will tell you. You will see that it gives you a vertex cover. And again, you can have a fractional vertex cover, but a very similar argument will tell you that you can find an optimal vertex cover which is an integer solution to the linear program. And that shows Konig's theorem.

And we will be doing things that are more like this, taking linear programs for interesting problems, and we will do this with zero sum games and maximum flow, and show first that the fractional solutions can always be massaged to give you integer solutions for maximum flow. And this will give us a duality theorem that the maximum flow equals the minimum cut.

And for zero sum games, a game is a two player things, and we will show that the primal linear program gives you the optimal strategy for the first player, and the dual linear program gives you the optimal strategy for the second player. And because they are duals, they have the same objective function. And that tells us that if the first player plays his optimum strategy, he can always guarantee a win of, say, w on each turn on average.

And if the second player plays his optimal strategy, he can always achieve a loss of at most w on each turn. So the first player and the second player, first player can guarantee he wins plus w, and the second player can guarantee he loses no more than minus w. And so this is the value of the game.