

# **ORF307 – Optimization**

## **17. Interior-point methods**

**Recap**

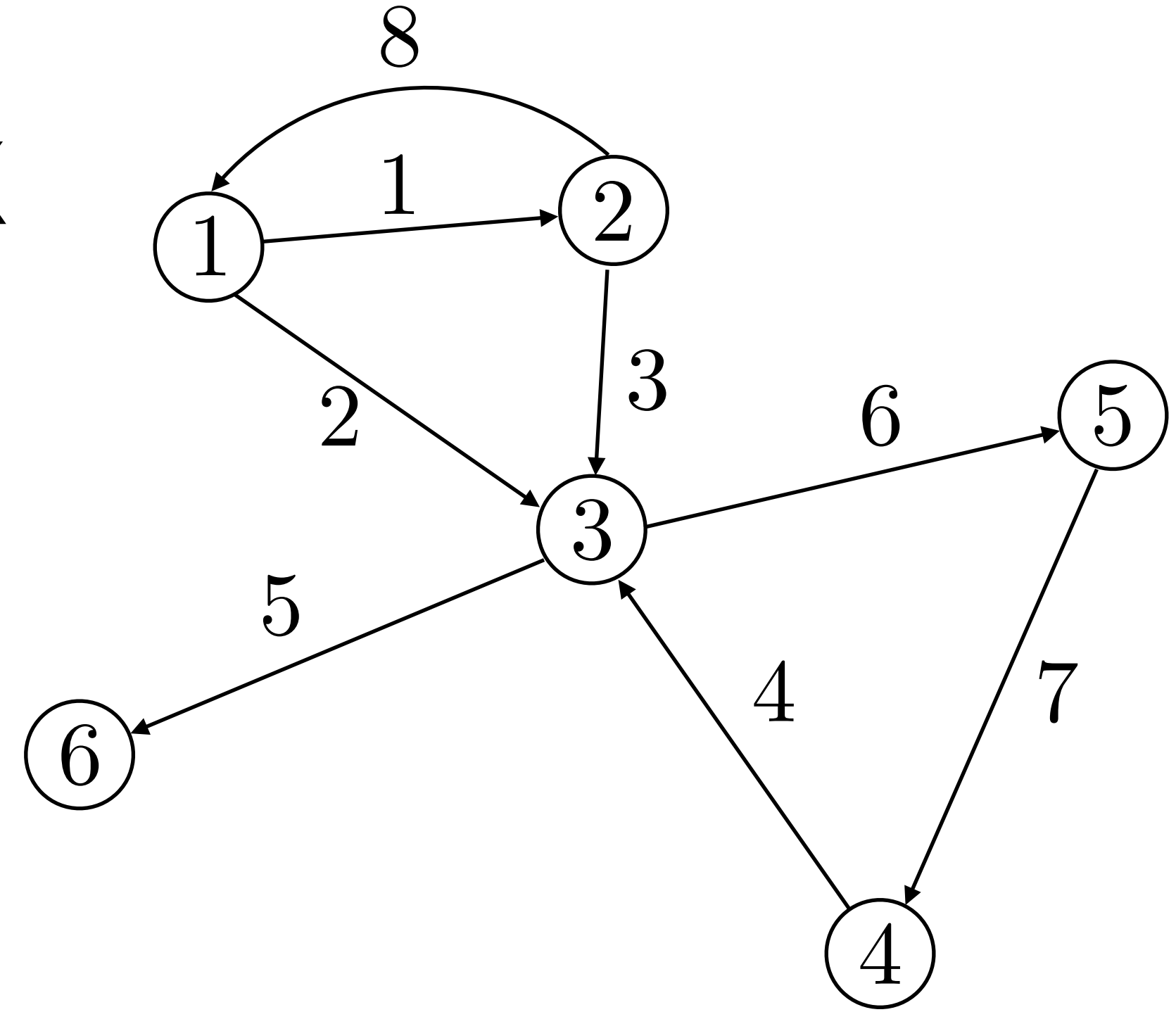
# Arc-node incidence matrix

$m \times n$  matrix  $A$  with entries

$$A_{ij} = \begin{cases} 1 & \text{if arc } j \text{ starts at node } i \\ -1 & \text{if arc } j \text{ ends at node } i \\ 0 & \text{otherwise} \end{cases}$$

**Note** Each column has  
one  $-1$  and one  $1$

# Arc-node incidence matrix



$m \times n$  matrix  $A$  with entries

$$A_{ij} = \begin{cases} 1 & \text{if arc } j \text{ starts at node } i \\ -1 & \text{if arc } j \text{ ends at node } i \\ 0 & \text{otherwise} \end{cases}$$

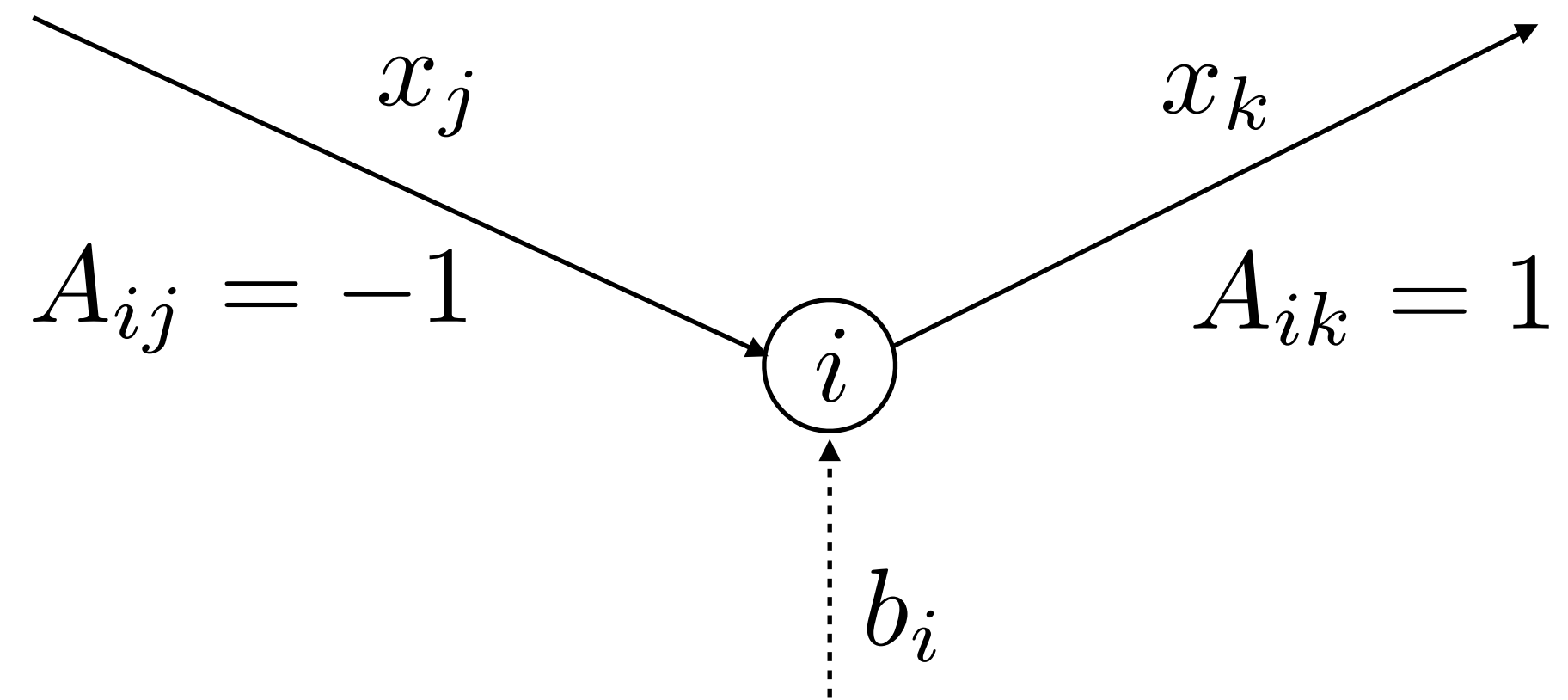
**Note** Each column has one  $-1$  and one  $1$

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & -1 & -1 & -1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \end{bmatrix}$$

# External supply

supply vector  $b \in \mathbb{R}^m$

- $b_i$  is the external supply at node  $i$   
(if  $b_i < 0$ , it represents demand)
- We must have  $\mathbf{1}^T b = 0$   
(total supply = total demand)



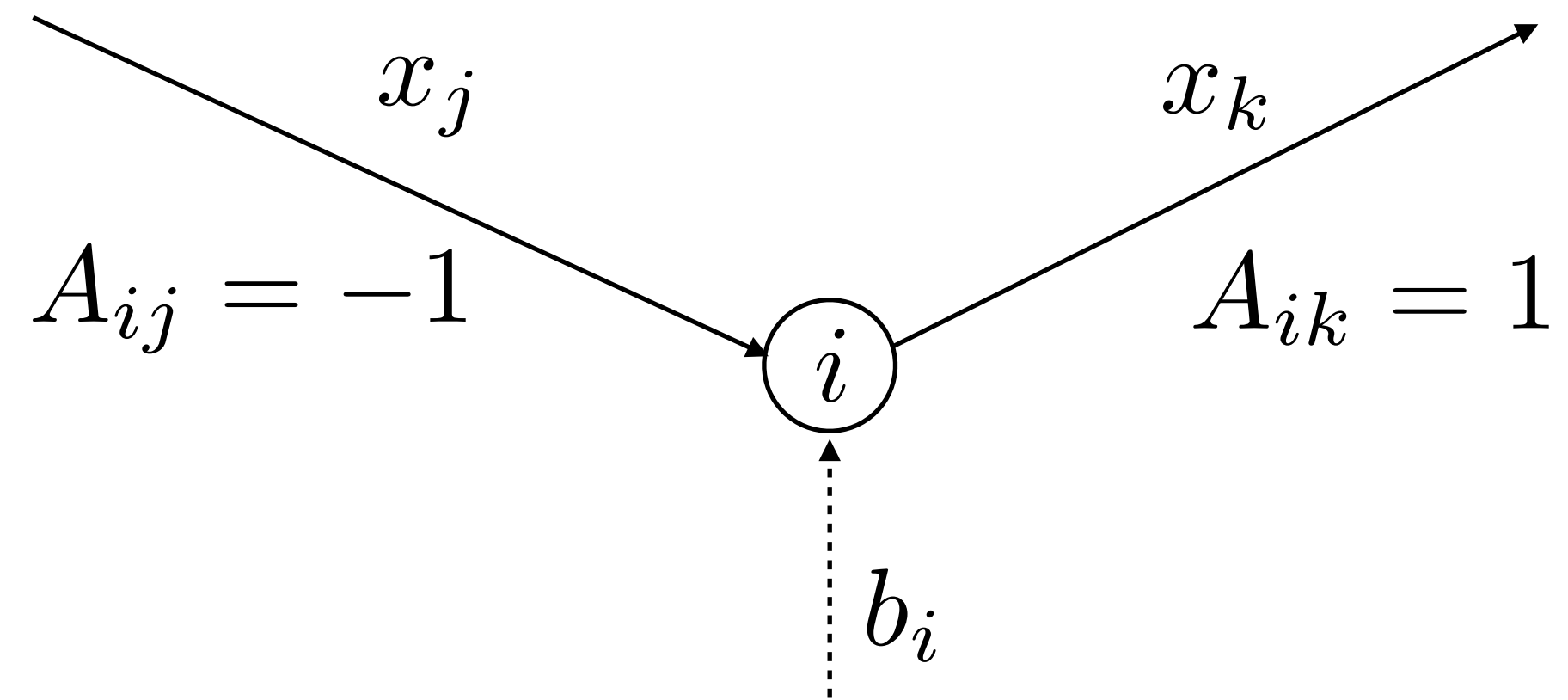
# External supply

supply vector  $b \in \mathbb{R}^m$

- $b_i$  is the external supply at node  $i$  (if  $b_i < 0$ , it represents demand)
- We must have  $\mathbf{1}^T b = 0$  (total supply = total demand)

## Balance equations

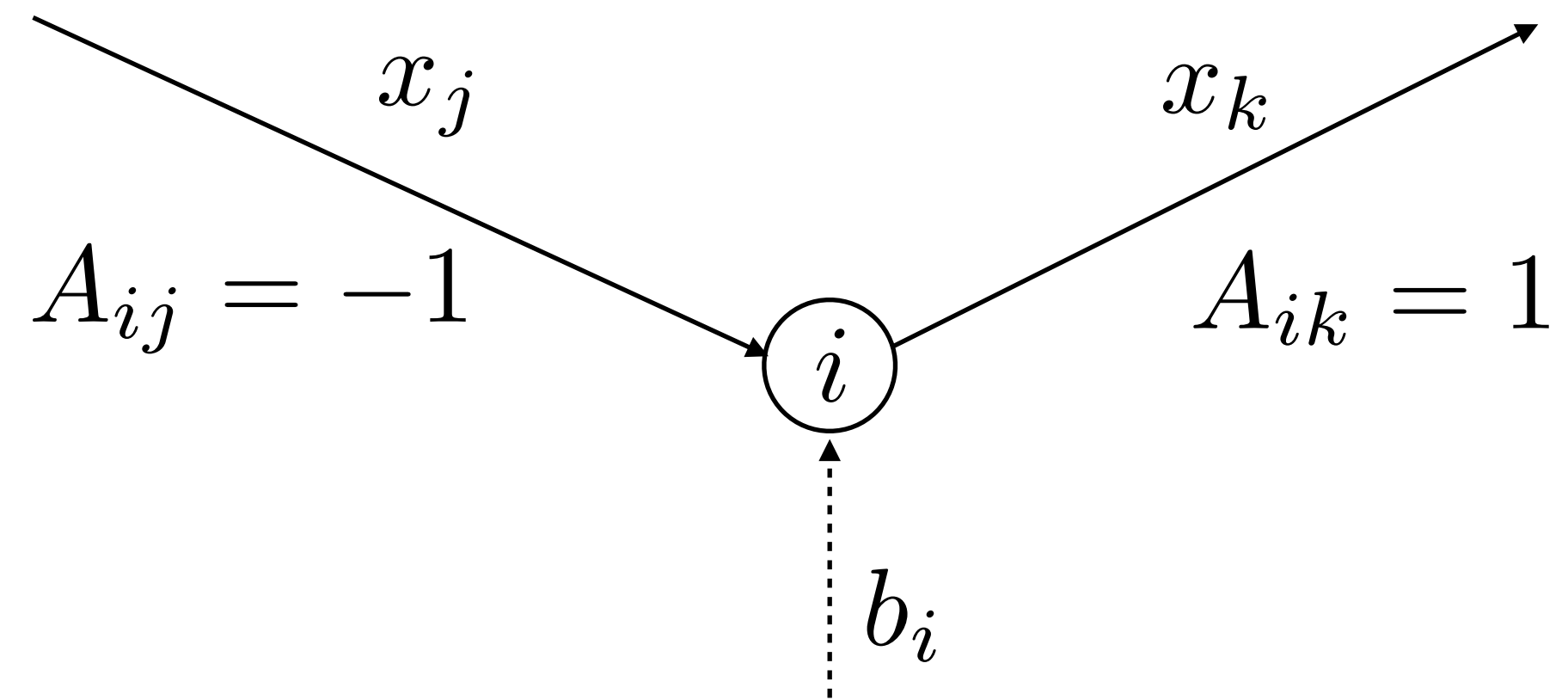
$$\sum_{j=1}^n A_{ij} x_j = (Ax)_i = b_i, \quad \text{for all } i$$



# External supply

supply vector  $b \in \mathbb{R}^m$

- $b_i$  is the external supply at node  $i$  (if  $b_i < 0$ , it represents demand)
- We must have  $\mathbf{1}^T b = 0$  (total supply = total demand)



## Balance equations

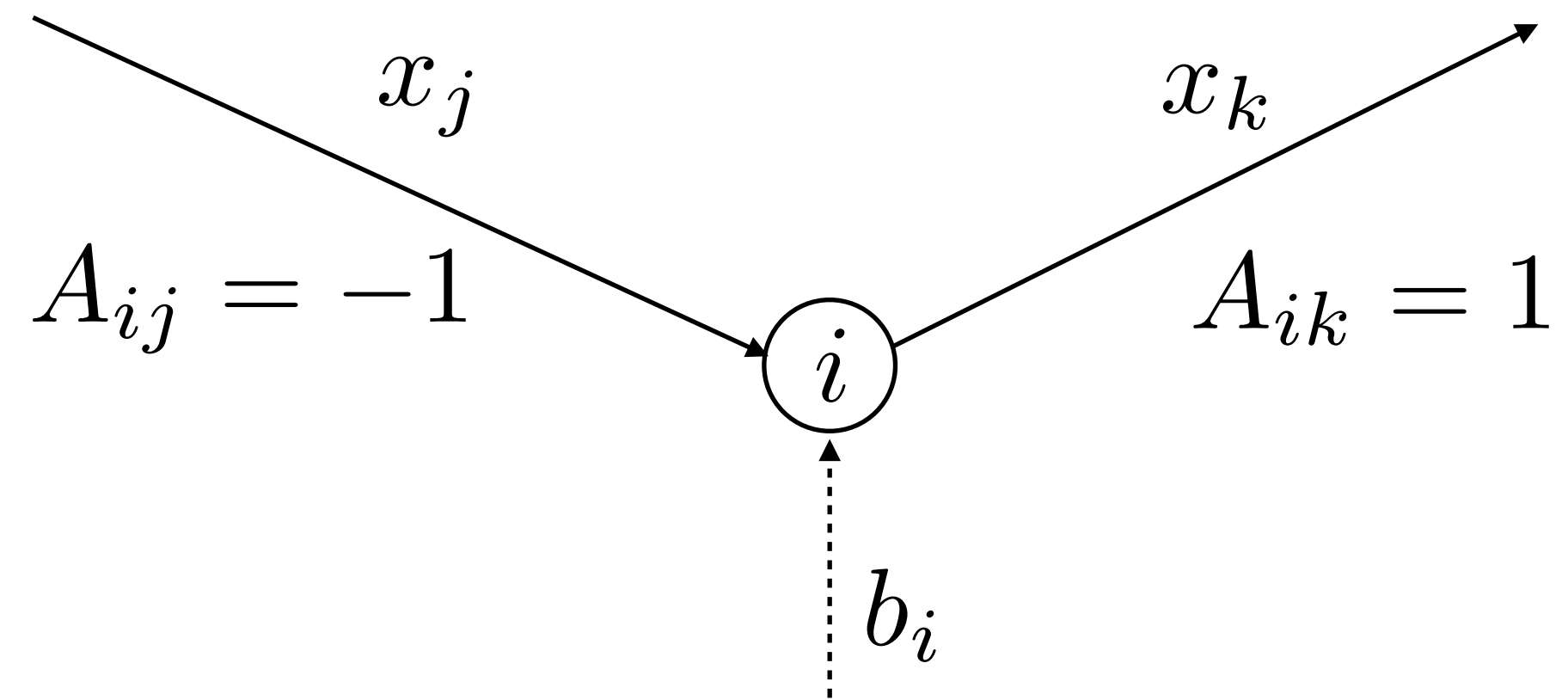
$$\sum_{j=1}^n A_{ij} x_j = (Ax)_i = b_i, \quad \text{for all } i$$

Total leaving  
flow

# External supply

supply vector  $b \in \mathbb{R}^m$

- $b_i$  is the external supply at node  $i$  (if  $b_i < 0$ , it represents demand)
- We must have  $\mathbf{1}^T b = 0$  (total supply = total demand)



## Balance equations

$$\sum_{j=1}^n A_{ij} x_j = (Ax)_i = b_i, \quad \text{for all } i$$

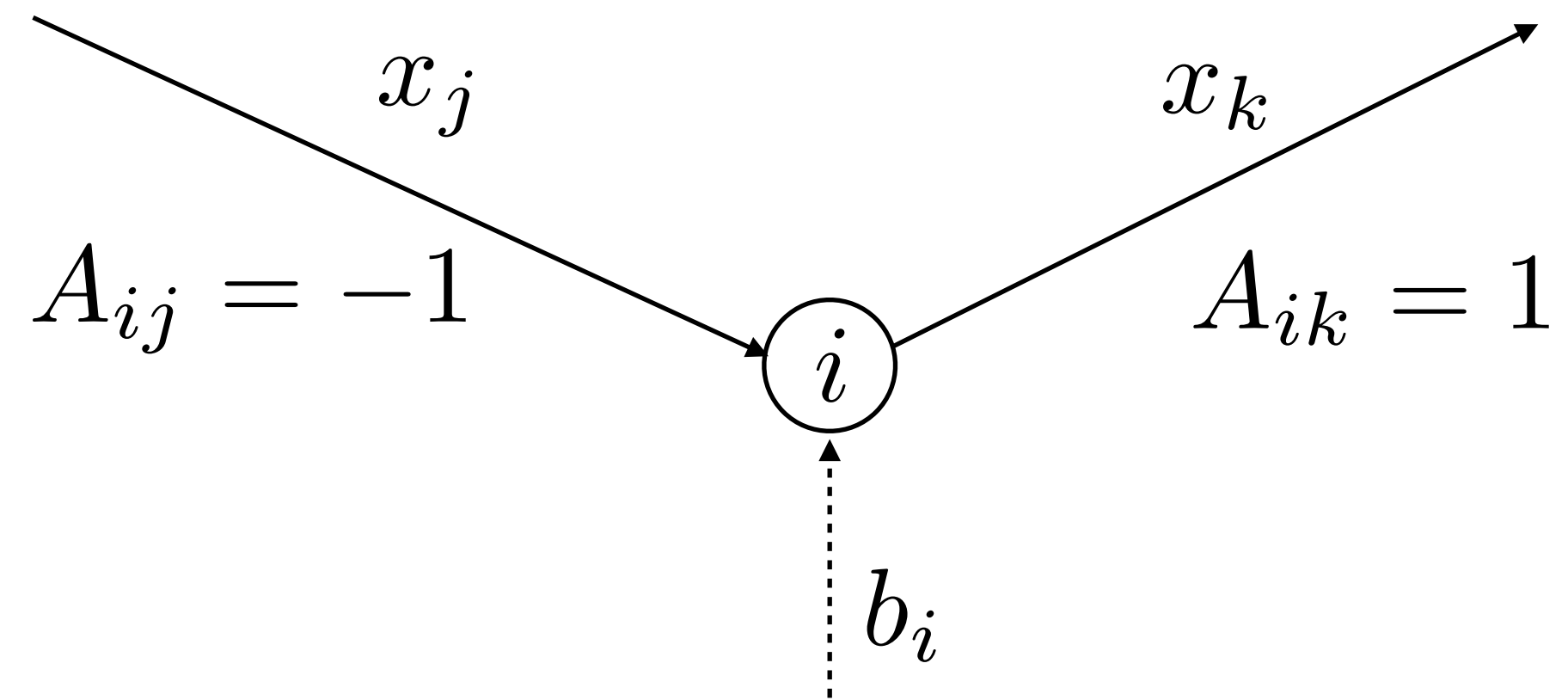
Total leaving flow                  Supply



# External supply

supply vector  $b \in \mathbb{R}^m$

- $b_i$  is the external supply at node  $i$  (if  $b_i < 0$ , it represents demand)
- We must have  $\mathbf{1}^T b = 0$  (total supply = total demand)



## Balance equations

$$\sum_{j=1}^n A_{ij} x_j = (Ax)_i = b_i, \quad \text{for all } i$$

Total leaving  
flow

Supply



$$Ax = b$$

# Minimum cost network flow problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & 0 \leq x \leq u \end{array}$$

- $c_i$  is unit cost of flow through arc  $i$
- Flow  $x_i$  must be nonnegative
- $u_i$  is the maximum flow capacity of arc  $i$
- Many network optimization problems are just special cases



# Integrality theorem

Given a polyhedron  $P = \{x \in \mathbf{R}^n \mid Ax = b, \quad x \geq 0\}$

where

- $A$  is totally unimodular
  - $b$  is an integer vector
- 
- all the extreme points of  $P$  are integer vectors.

## Proof

- All extreme points are basic feasible solutions with  $x_B = A_B^{-1}b$  and  $x_i = 0, i \neq B$
- $A_B^{-1}$  has integer components because of total unimodularity of  $A$
- $b$  has also integer components
- Therefore, also  $x$  is integral



# Implications for network and combinatorial optimization

## Minimum cost network flow

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & 0 \leq x \leq u \end{array}$$



If  $b$  and  $u$  are integral solutions  $x^*$  are integral

# Implications for network and combinatorial optimization

## Minimum cost network flow

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & 0 \leq x \leq u \end{array}$$



If  $b$  and  $u$  are integral solutions  $x^*$  are integral

## Integer linear programs

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & 0 \leq x \leq u \\ & x \in \mathbf{Z}^n \end{array}$$

Very difficult in general  
(more on this in a few weeks)

# Implications for network and combinatorial optimization

## Minimum cost network flow

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & 0 \leq x \leq u \end{array}$$



If  $b$  and  $u$  are integral solutions  $x^*$  are integral

## Integer linear programs

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & 0 \leq x \leq u \\ & x \in \mathbf{Z}^n \end{array}$$

Very difficult in general  
(more on this in a few weeks)



If  $A$  totally unimodular and  $b, u$  integral, we can relax integrality and solve a fast LP instead

# Today's lecture

## Interior point methods

- History
- Newton's method
- Central path
- Primal-dual path-following algorithm
- Logarithmic barrier functions



# History

# A brief history of linear optimization

## 1940s :

- Foundations and applications in economics and logistics (Kantorovich, Koopmans)
- **1947** : Development of the **simplex method** by Dantzig

## 1950s – 70s:

- Applications expand to engineering, OR, computer science...
- **1975** : Nobel prize in economics for Kantorovich and Koopmans

## 1980s:

- Development of polynomial time algorithms for LPs
- **1984** : Development of the **interior point method** by Karmarkar

## —Today:

- Continued algorithm development. Expansion to very large problems.

# Ellipsoid method Khachian (1979)

Answer to major question  
Is worst-case LP complexity  
polynomial? **Yes!**

## Shazam! A Shortcut for Computers

*A garment manufacturer has three kinds of dresses — A, B and C. On hand he has 17 bolts of one cloth and 25 of another, as well as 200 buttons and 75 belts. He has three cutters, 10 sewers and one finisher. Dress A, on which he makes a profit of \$1.25 a unit, requires one combination of material, accessories and work; the B dress, with a \$1.50 profit, takes a different combination, and the \$2.25 dress C has yet a third set of requirements. How should he schedule his production to make the most money?*

That is an easy example of a kind of eminently practical problem that becomes computationally difficult because of the number of variable factors and constraints that must be handled to get a best solution. And, as the number of variables and restraints grows — as, for instance, in a model of the national economy or in the scheduling of production at any oil refinery — the difficulty mushrooms.

Even the most powerful computers might have to run for hours to tell a plant manager how to handle a small change in, say, the amount of crude oil being delivered to his tanks. And adding one new restriction can substantially increase the number of possible answers and thus the time required to check them for an optimum solution.

Last week, intrigued mathematicians were trying to sort out the meaning of what looked like a stun-

ning theoretical breakthrough in the handling of these “linear programming” problems — and some were wondering why it had taken so long for the breakthrough to become generally known.

In January, the Soviet journal *Doklady* published an abstract of the new solution put forward by a Russian mathematician, L. G. Khachian, about whom no further biographical data has been made public. The abstract was generally overlooked until two mathematicians, working at Stanford University, analyzed the theory and refined its application. Reports of their work and Mr. (or Miss) Khachian’s began appearing in American journals four weeks ago, opening up the floodgates of mathematical curiosity.

Ronald L. Graham, a leading computer expert at the Bell Laboratories in Murray Hill, N.J., said the significance of the new method is that it provides a fast way to test whether there is an optimum solution for any particular linear programming problem and, if there is, to assure that the solution can be computed within a reasonable length of time.

The older, “simplex” method involved having the computer “build” a flat-sided polyhedron in multidimensional space and then hop from vertex to vertex testing for a best answer. Mr. Khachian’s solution has the computer design a multidimensional curved ellipsoid that sur-

rounds the area of possible solutions and is then made smaller until it neatly encloses the optimum answer.

The practical effects of the breakthrough were not entirely clear last week, however. Although it seems to offer enormous advantages in areas ranging from industrial scheduling to weather forecasting, it has yet to be tested in the development of an actual major computer program. Dr. Graham said it might work for some kinds of linear programming problems and not for others, noting that, despite its theoretical limitations, simplex in fact works quite efficiently for the problems it has been asked to handle.

Nevertheless Laslo Lovász, a Hungarian mathematician who worked on the problem at Stanford, said he used the method to program his pocket calculator to solve a problem with six variables and six constraints, which it probably could not have handled with the simplex method. And George B. Dantzig, who devised the simplex method in 1947, said he felt “stupid that I didn’t see” Mr. Khachian’s method.

While some wondered about the delay between the publication of the abstract in *Doklady* and its reception now, others pointed out that “simplex” itself it did not get into wide use until several years after its theoretical formulation.

— JONATHAN FRIENDLY

*The New York Times*

Published: November 11, 1979

# Ellipsoid method

## Khachian (1979)

### Answer to major question

Is worst-case LP complexity polynomial? **Yes!**

### Drawbacks

Very inefficient. Much slower than simplex!

## Shazam! A Shortcut for Computers

*A garment manufacturer has three kinds of dresses — A, B and C. On hand he has 17 bolts of one cloth and 25 of another, as well as 200 buttons and 75 belts. He has three cutters, 10 sewers and one finisher. Dress A, on which he makes a profit of \$1.25 a unit, requires one combination of material, accessories and work; the B dress, with a \$1.50 profit, takes a different combination, and the \$2.25 dress C has yet a third set of requirements. How should he schedule his production to make the most money?*

That is an easy example of a kind of eminently practical problem that becomes computationally difficult because of the number of variable factors and constraints that must be handled to get a best solution. And, as the number of variables and restraints grows — as, for instance, in a model of the national economy or in the scheduling of production at any oil refinery — the difficulty mushrooms.

Even the most powerful computers might have to run for hours to tell a plant manager how to handle a small change in, say, the amount of crude oil being delivered to his tanks. And adding one new restriction can substantially increase the number of possible answers and thus the time required to check them for an optimum solution.

Last week, intrigued mathematicians were trying to sort out the meaning of what looked like a stun-

ning theoretical breakthrough in the handling of these “linear programming” problems — and some were wondering why it had taken so long for the breakthrough to become generally known.

In January, the Soviet journal *Doklady* published an abstract of the new solution put forward by a Russian mathematician, L. G. Khachian, about whom no further biographical data has been made public. The abstract was generally overlooked until two mathematicians, working at Stanford University, analyzed the theory and refined its application. Reports of their work and Mr. (or Miss) Khachian’s began appearing in American journals four weeks ago, opening up the floodgates of mathematical curiosity.

Ronald L. Graham, a leading computer expert at the Bell Laboratories in Murray Hill, N.J., said the significance of the new method is that it provides a fast way to test whether there is an optimum solution for any particular linear programming problem and, if there is, to assure that the solution can be computed within a reasonable length of time.

The older, “simplex” method involved having the computer “build” a flat-sided polyhedron in multidimensional space and then hop from vertex to vertex testing for a best answer. Mr. Khachian’s solution has the computer design a multidimensional curved ellipsoid that sur-

rounds the area of possible solutions and is then made smaller until it neatly encloses the optimum answer.

The practical effects of the breakthrough were not entirely clear last week, however. Although it seems to offer enormous advantages in areas ranging from industrial scheduling to weather forecasting, it has yet to be tested in the development of an actual major computer program. Dr. Graham said it might work for some kinds of linear programming problems and not for others, noting that, despite its theoretical limitations, simplex in fact works quite efficiently for the problems it has been asked to handle.

Nevertheless Laslo Lovász, a Hungarian mathematician who worked on the problem at Stanford, said he used the method to program his pocket calculator to solve a problem with six variables and six constraints, which it probably could not have handled with the simplex method. And George B. Dantzig, who devised the simplex method in 1947, said he felt “stupid that I didn’t see” Mr. Khachian’s method.

While some wondered about the delay between the publication of the abstract in *Doklady* and its reception now, others pointed out that “simplex” itself it did not get into wide use until several years after its theoretical formulation.

— JONATHAN FRIENDLY

*The New York Times*

Published: November 11, 1979

# Ellipsoid method Khachian (1979)

## Answer to major question

Is worst-case LP complexity polynomial? **Yes!**

## Drawbacks

Very inefficient. Much slower than simplex!

## Benefits

Motivated new research directions

## Shazam! A Shortcut for Computers

*A garment manufacturer has three kinds of dresses — A, B and C. On hand he has 17 bolts of one cloth and 25 of another, as well as 200 buttons and 75 belts. He has three cutters, 10 sewers and one finisher. Dress A, on which he makes a profit of \$1.25 a unit, requires one combination of material, accessories and work; the B dress, with a \$1.50 profit, takes a different combination, and the \$2.25 dress C has yet a third set of requirements. How should he schedule his production to make the most money?*

That is an easy example of a kind of eminently practical problem that becomes computationally difficult because of the number of variable factors and constraints that must be handled to get a best solution. And, as the number of variables and restraints grows — as, for instance, in a model of the national economy or in the scheduling of production at any oil refinery — the difficulty mushrooms.

Even the most powerful computers might have to run for hours to tell a plant manager how to handle a small change in, say, the amount of crude oil being delivered to his tanks. And adding one new restriction can substantially increase the number of possible answers and thus the time required to check them for an optimum solution.

Last week, intrigued mathematicians were trying to sort out the meaning of what looked like a stun-

ning theoretical breakthrough in the handling of these “linear programming” problems — and some were wondering why it had taken so long for the breakthrough to become generally known.

In January, the Soviet journal *Doklady* published an abstract of the new solution put forward by a Russian mathematician, L. G. Khachian, about whom no further biographical data has been made public. The abstract was generally overlooked until two mathematicians, working at Stanford University, analyzed the theory and refined its application. Reports of their work and Mr. (or Miss) Khachian’s began appearing in American journals four weeks ago, opening up the floodgates of mathematical curiosity.

Ronald L. Graham, a leading computer expert at the Bell Laboratories in Murray Hill, N.J., said the significance of the new method is that it provides a fast way to test whether there is an optimum solution for any particular linear programming problem and, if there is, to assure that the solution can be computed within a reasonable length of time.

The older, “simplex” method involved having the computer “build” a flat-sided polyhedron in multidimensional space and then hop from vertex to vertex testing for a best answer. Mr. Khachian’s solution has the computer design a multidimensional curved ellipsoid that sur-

rounds the area of possible solutions and is then made smaller until it neatly encloses the optimum answer.

The practical effects of the breakthrough were not entirely clear last week, however. Although it seems to offer enormous advantages in areas ranging from industrial scheduling to weather forecasting, it has yet to be tested in the development of an actual major computer program. Dr. Graham said it might work for some kinds of linear programming problems and not for others, noting that, despite its theoretical limitations, simplex in fact works quite efficiently for the problems it has been asked to handle.

Nevertheless Laslo Lovász, a Hungarian mathematician who worked on the problem at Stanford, said he used the method to program his pocket calculator to solve a problem with six variables and six constraints, which it probably could not have handled with the simplex method. And George B. Dantzig, who devised the simplex method in 1947, said he felt “stupid that I didn’t see” Mr. Khachian’s method.

While some wondered about the delay between the publication of the abstract in *Doklady* and its reception now, others pointed out that “simplex” itself it did not get into wide use until several years after its theoretical formulation.

— JONATHAN FRIENDLY

*The New York Times*

Published: November 11, 1979

# Interior-point methods

## 1980s-1990s: interior point methods

- Karmarkar's algorithm (1984)
- Competitive with simplex, often faster for larger problems
- Began huge effort in algorithm development for convex optimization



**SOVIETS HELD TO DELIVER U.S. FAMINE SUPPLIES** In Kambelaha, Ethiopia, west from the United States, a Soviet helicopter is hoisted onto a Soviet helicopter to help relieve emergency. Page A12.

### Cocaine Traffickers Kill 17 in Peru Raid On Antidrug Team

LIMA, Peru, Nov. 18 — A band of cocaine traffickers burst into a jungle campsite and opened fire with machine guns, killing at least 17 people employed by a United States-financed program to destroy coca crops, the police said today.

All those killed in the attack, which took place early Saturday, were identified as Peruvian employees of the Coca Reduction Organization. The group is taking part in a \$20 million program that the United States is financing to cut the production of coca along the Huastilla River, west more than the illegal coca in Peru's grown.

Substrate crops offered Peru products about half of the world's coca, the prime ingredient in cocaine. Workers in the program destroy coca and spend five with machine guns, killing at least 17 people employed by a United States-financed program to destroy coca crops, the police said today.

All those killed in the attack, which took place early Saturday, were identified as Peruvian employees of the Coca Reduction Organization. The group is taking part in a \$20 million program that the United States is financing to cut the production of coca along the Huastilla River, west more than the illegal coca in Peru's grown.

Under suspicious circumstances, the group was killed by a band of cocaine traffickers. The group was identified as Peruvian employees of the Coca Reduction Organization. The group is taking part in a \$20 million program that the United States is financing to cut the production of coca along the Huastilla River, west more than the illegal coca in Peru's grown.

Substrate crops offered Peru products about half of the world's coca, the prime ingredient in cocaine. Workers in the program destroy coca and spend five with machine guns, killing at least 17 people employed by a United States-financed program to destroy coca crops, the police said today.

### Vote Comes to a 'Homeland,' But African Problems Linger

Under suspicious circumstances, the group was killed by a band of cocaine traffickers. The group was identified as Peruvian employees of the Coca Reduction Organization. The group is taking part in a \$20 million program that the United States is financing to cut the production of coca along the Huastilla River, west more than the illegal coca in Peru's grown.

Substrate crops offered Peru products about half of the world's coca, the prime ingredient in cocaine. Workers in the program destroy coca and spend five with machine guns, killing at least 17 people employed by a United States-financed program to destroy coca crops, the police said today.

### Breakthrough in Problem Solving

A 28-year-old mathematician at A.T.&T. Bell Laboratories has made a startling theoretical breakthrough in the solving of systems of equations that often grow too vast and complex for the most powerful computers.

The discovery, which is to be formally published next month, is already circulating rapidly through the mathematical world. It has also set off a deluge of inquiries from brokerage houses, oil companies and airlines, industries with millions of dollars at stake in problems known as linear programming.

These problems are fiendishly complicated systems, often with thousands of variables. They arise in a variety of commercial and government applications, ranging from allocating time on a communications satellite to routing millions of telephone calls over long distances. Linear programming is particularly useful whenever a limited, expensive resource must be spread most efficiently among competing users. And investment companies use the approach in creating portfolios with the best mix of stocks and bonds.

Faster Solutions Seen The Bell Labs mathematician, Dr. Narendra Karmarkar, has devised a radically new procedure that may speed the routine handling of such problems by businesses and Government agencies and also make it possible to tackle problems that are now far out of reach.

"This is a path-breaking result," said Dr. Ronald L. Graham, director of mathematical sciences for Bell Labs in Murray Hill, N.J. "Science has its mo-

### AMERICANS IN POLL VIEW GOVERNMENT MORE CONFIDENTLY

But Postelection Inquiry Also Finds Most Think Reagan Will Ask Rise in Taxes

The American public, its level of confidence in government rebounding after more than a decade of doubt, expects President Reagan to avoid an economic recession in his second term and to make a real effort to negotiate an arms control treaty, a New York Times survey of public opinion found.

But at the same time, the public expects him to break his most ambitious campaign promise and ask Congress to vote an increase in taxes. Fifty-seven percent of the public and 40 percent of voters expect him to ask for higher taxes.

### Albany Leaders Predicting a Cut In Income Taxes

Republicans and Democrats in Albany predicted yesterday that a cut in personal income taxes similar to that recommended by a Cuomo Administration panel over low revenues seems to be the way to go.

At the same time, however, Republicans said they have had to drop Governor Cuomo "screaming and kicking" into the process. Democrats said the panel's report merely reflected a common sense developed in the Legislature.

The citizens panel, comprising 20 business and civic leaders appointed by Mr. Cuomo, said personal income taxes should be cut by \$15 million a year. This would lower taxes by about 6 percent for most people earning more than \$15,000 and up to 24 percent for those earning less.

Single-Digit Top Rate Under the plan, the state's maximum tax rate on married income would drop by 1 percentage point, to 9 percent. In addition, the personal exemption would be raised to \$1,000 from \$600, which would help lower-income workers, and the standard deduction would be set at the Federal level of \$2,300 for a single person and \$2,400 for a married couple.

Even with Congress in adjournment and most legislators on vacation, the public has begun. Aides to several members of Congress reported stacks of telephone messages from business lobbyists who want to pressure the accelerated depreciation feature of the bill tax cut.

According to Treasury officials, Secretary Donald T. Regan said his staff Friday that he wanted to modify but not abolish that feature, which is called the accelerated cost recovery system.

The system allows large write-offs for companies that invest heavily in real estate, plants and machinery and is little use to many other companies. It is one of the most controversial tax cuts ever suggested by the Administration.

The centerpiece of the Administration's 1985 business tax cut, the depreciation system, is getting the Government's early 1985 budget in last year's revenue.

Mr. Cuomo's secretary, Michael J. DeGiulio, said that barring any startling change in the state's fiscal and economic condition, Mr. Cuomo would

For the last 10 weeks, homeless families, mostly mothers and young children, have been spending weekend nights on plastic chairs, on counter tops or on the floor in New York City's emergency welfare office because the city's welfare agency has run out of beds.

Other families have been waiting at most through the night while city welfare workers try to find temporary space for them in any of the 12 hotels scattered throughout Manhattan, the Bronx, Brooklyn and Queens that accept emergency welfare cases.

In some cases, the families leave the Manhattan office at 6 or even 5 A.M. for an hour's trip on the subway to hotels in the other boroughs that will receive them to check out as early as 11 A.M. the same morning.

Struggling to Meet Need City officials acknowledged the problem yesterday, and said they were increasing to keep pace with the ever-increasing need for emergency and permanent housing for poor families. This weekend the city opened three additional emergency welfare offices — one each in Brooklyn, Queens and the Bronx — to relieve the pressure at the office at 241 Church Street in lower Manhattan.

As of Oct. 31, 1,176 families who can no longer stay with relatives, or who have been evicted, or whose apartment buildings have been closed down, or who have lost their homes for a variety of other reasons were being sheltered by the city in hotels or in its four shelters for families. That is 1,000 more families than a year ago, according to the city's Human Resources Administration.

Homeless people waiting at the Emergency Assistance Unit at 241 Church Street at city welfare workers try to find temporary space for them.

space at our shelters, there is no way for us to invest space," Mr. Deacy said yesterday. "But tonight, we're able to say that this is a dramatic situation," said Stanley Brumfield, the Deputy Commissioner of the Department of Social Services.

people are coming to us for temporary space for them. There are available apartments for them. "I don't think anyone would want to stay out here in a cardboard situation," said Stanley Brumfield, the Deputy Commissioner of the Department of Social Services.

### Breakthrough in Problem Solving

ments of great progress, and this may well be one of them."

Because problems in linear programming can have billions or more possible answers, even high-speed computers cannot check every one. So computers must use a special procedure, an algorithm, to examine as few answers as possible before finding the best one — typically the one that minimizes cost or maximizes efficiency.

A procedure devised in 1947, the simplex method, is now used for such problems of great progress, and this may well be one of them."

These problems are fiendishly complicated systems, often with thousands of variables. They arise in a variety of commercial and government applications, ranging from allocating time on a communications satellite to routing millions of telephone calls over long distances. Linear programming is particularly useful whenever a limited, expensive resource must be spread most efficiently among competing users. And investment companies use the approach in creating portfolios with the best mix of stocks and bonds.

Faster Solutions Seen The Bell Labs mathematician, Dr. Narendra Karmarkar, has devised a radically new procedure that may speed the routine handling of such problems by businesses and Government agencies and also make it possible to tackle problems that are now far out of reach.

"This is a path-breaking result," said Dr. Ronald L. Graham, director of mathematical sciences for Bell Labs in Murray Hill, N.J. "Science has its mo-

ments of great progress, and this may well be one of them."

These problems are fiendishly complicated systems, often with thousands of variables. They arise in a variety of commercial and government applications, ranging from allocating time on a communications satellite to routing millions of telephone calls over long distances. Linear programming is particularly useful whenever a limited, expensive resource must be spread most efficiently among competing users. And investment companies use the approach in creating portfolios with the best mix of stocks and bonds.

Faster Solutions Seen The Bell Labs mathematician, Dr. Narendra Karmarkar, has devised a radically new procedure that may speed the routine handling of such problems by businesses and Government agencies and also make it possible to tackle problems that are now far out of reach.

"This is a path-breaking result," said Dr. Ronald L. Graham, director of mathematical sciences for Bell Labs in Murray Hill, N.J. "Science has its mo-

ments of great progress, and this may well be one of them."

These problems are fiendishly complicated systems, often with thousands of variables. They arise in a variety of commercial and government applications, ranging from allocating time on a communications satellite to routing millions of telephone calls over long distances. Linear programming is particularly useful whenever a limited, expensive resource must be spread most efficiently among competing users. And investment companies use the approach in creating portfolios with the best mix of stocks and bonds.

Faster Solutions Seen The Bell Labs mathematician, Dr. Narendra Karmarkar, has devised a radically new procedure that may speed the routine handling of such problems by businesses and Government agencies and also make it possible to tackle problems that are now far out of reach.

### Homeless Spend Nights in City Welfare Office

For the last 10 weeks, homeless families, mostly mothers and young children, have been spending weekend nights on plastic chairs, on counter tops or on the floor in New York City's emergency welfare office because the city's welfare agency has run out of beds.

Other families have been waiting at most through the night while city welfare workers try to find temporary space for them in any of the 12 hotels scattered throughout Manhattan, the Bronx, Brooklyn and Queens that accept emergency welfare cases.

In some cases, the families leave the Manhattan office at 6 or even 5 A.M. for an hour's trip on the subway to hotels in the other boroughs that will receive them to check out as early as 11 A.M. the same morning.

Struggling to Meet Need City officials acknowledged the problem yesterday, and said they were increasing to keep pace with the ever-increasing need for emergency and permanent housing for poor families. This weekend the city opened three additional emergency welfare offices — one each in Brooklyn, Queens and the Bronx — to relieve the pressure at the office at 241 Church Street in lower Manhattan.

As of Oct. 31, 1,176 families who can no longer stay with relatives, or who have been evicted, or whose apartment buildings have been closed down, or who have lost their homes for a variety of other reasons were being sheltered by the city in hotels or in its four shelters for families. That is 1,000 more families than a year ago, according to the city's Human Resources Administration.

Homeless people waiting at the Emergency Assistance Unit at 241 Church Street at city welfare workers try to find temporary space for them.

space at our shelters, there is no way for us to invest space," Mr. Deacy said yesterday. "But tonight, we're able to say that this is a dramatic situation," said Stanley Brumfield, the Deputy Commissioner of the Department of Social Services.

people are coming to us for temporary space for them. There are available apartments for them. "I don't think anyone would want to stay out here in a cardboard situation," said Stanley Brumfield, the Deputy Commissioner of the Department of Social Services.

ments of great progress, and this may well be one of them."

These problems are fiendishly complicated systems, often with thousands of variables. They arise in a variety of commercial and government applications, ranging from allocating time on a communications satellite to routing millions of telephone calls over long distances. Linear programming is particularly useful whenever a limited, expensive resource must be spread most efficiently among competing users. And investment companies use the approach in creating portfolios with the best mix of stocks and bonds.

Faster Solutions Seen The Bell Labs mathematician, Dr. Narendra Karmarkar, has devised a radically new procedure that may speed the routine handling of such problems by businesses and Government agencies and also make it possible to tackle problems that are now far out of reach.

"This is a path-breaking result," said Dr. Ronald L. Graham, director of mathematical sciences for Bell Labs in Murray Hill, N.J. "Science has its mo-

# Newton's method

# Newton's root finding method

**Goal:** solve

$$h(x) = 0$$



# Newton's root finding method

**Goal:** solve

$$h(x) = 0$$

## Method

1. Make a guess  $x^k$  and a linear approximation

$$h(x) \approx h(x^k) + \frac{\partial h}{\partial x^k}(x^k)(x - x^k)$$

2. Iteratively set  $\hat{h}(x)$  to 0

$$h(x^k) + \frac{\partial h}{\partial x^k}(x^k)(x^{k+1} - x^k) = 0$$

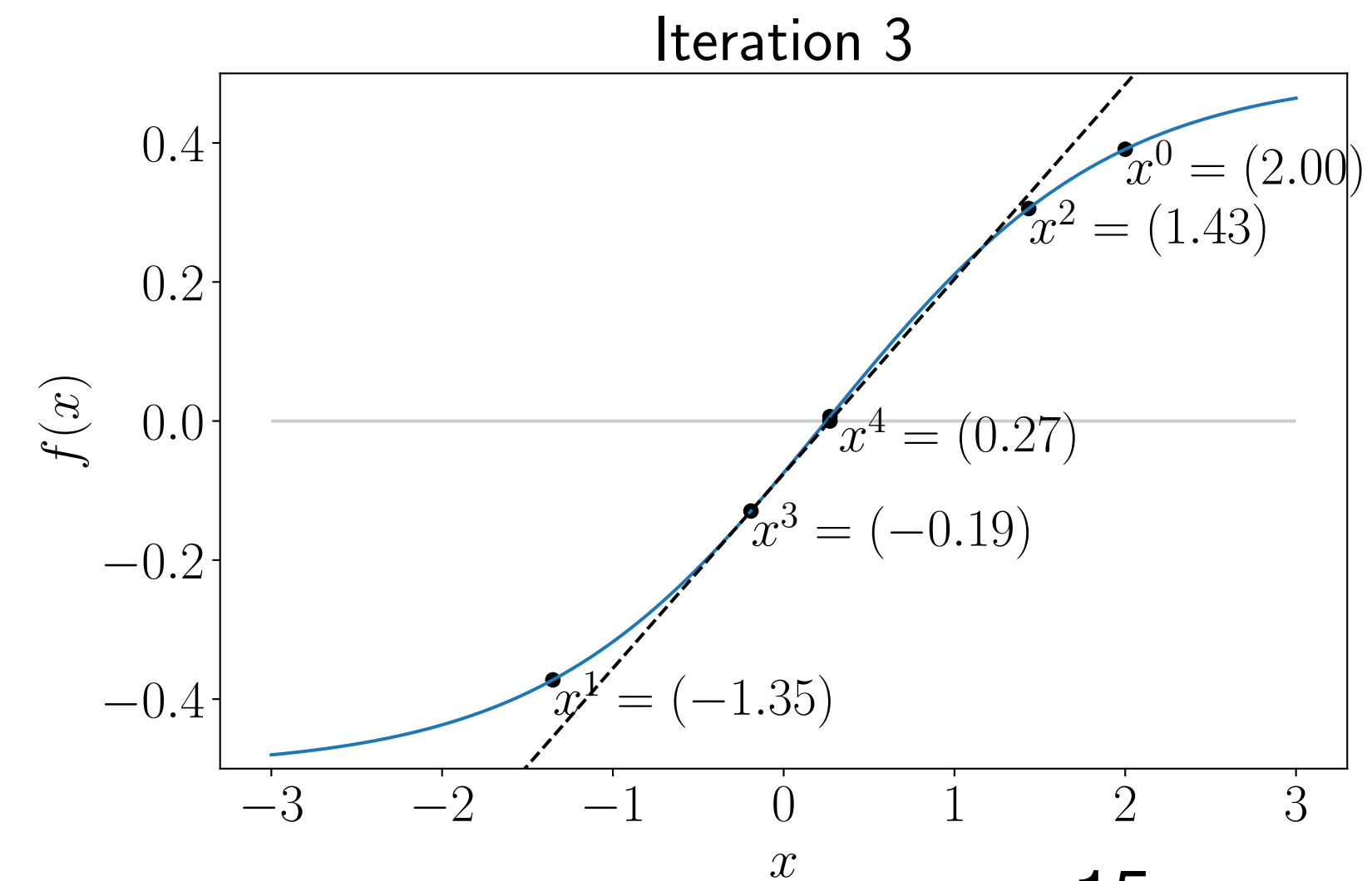
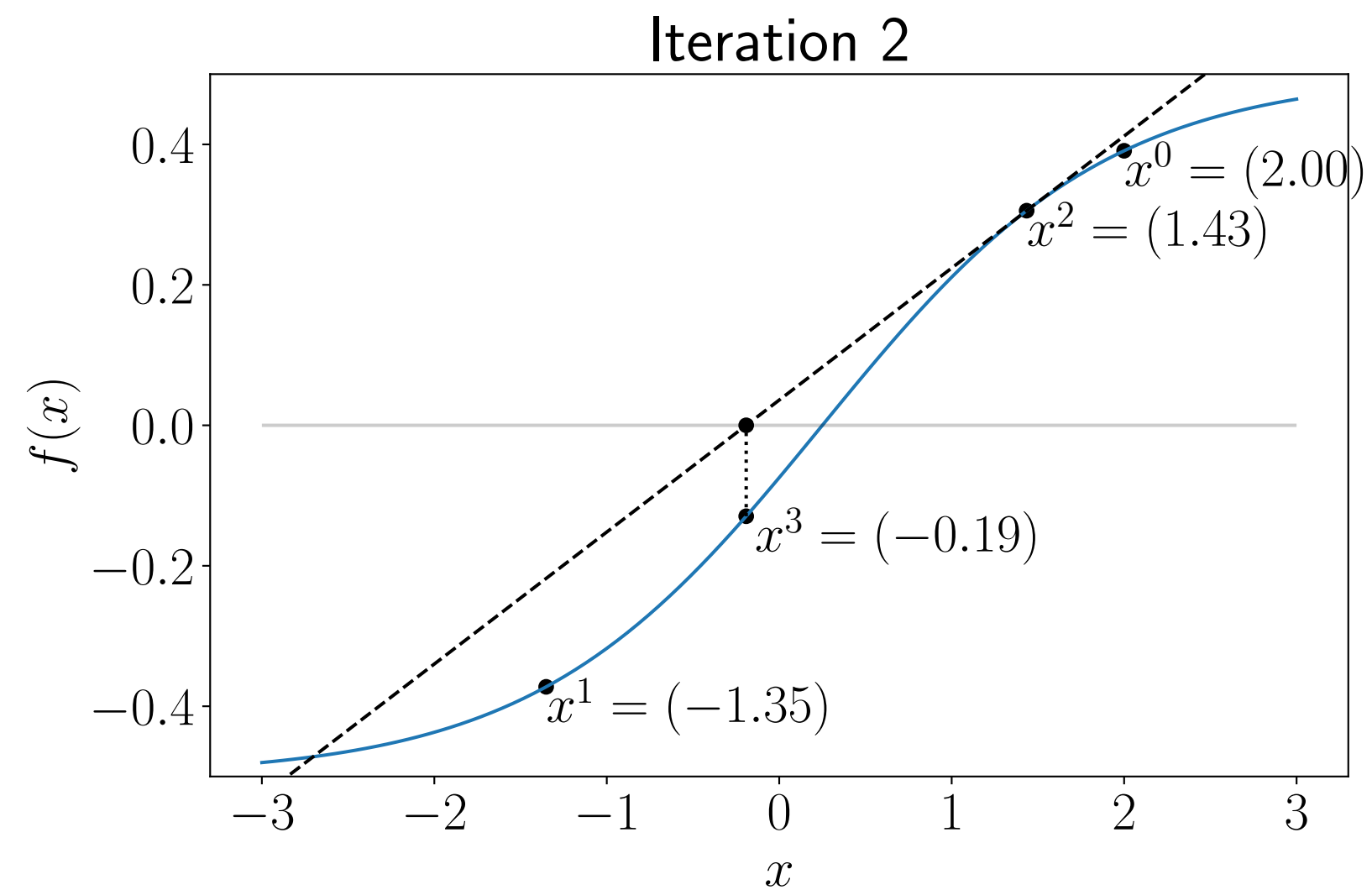
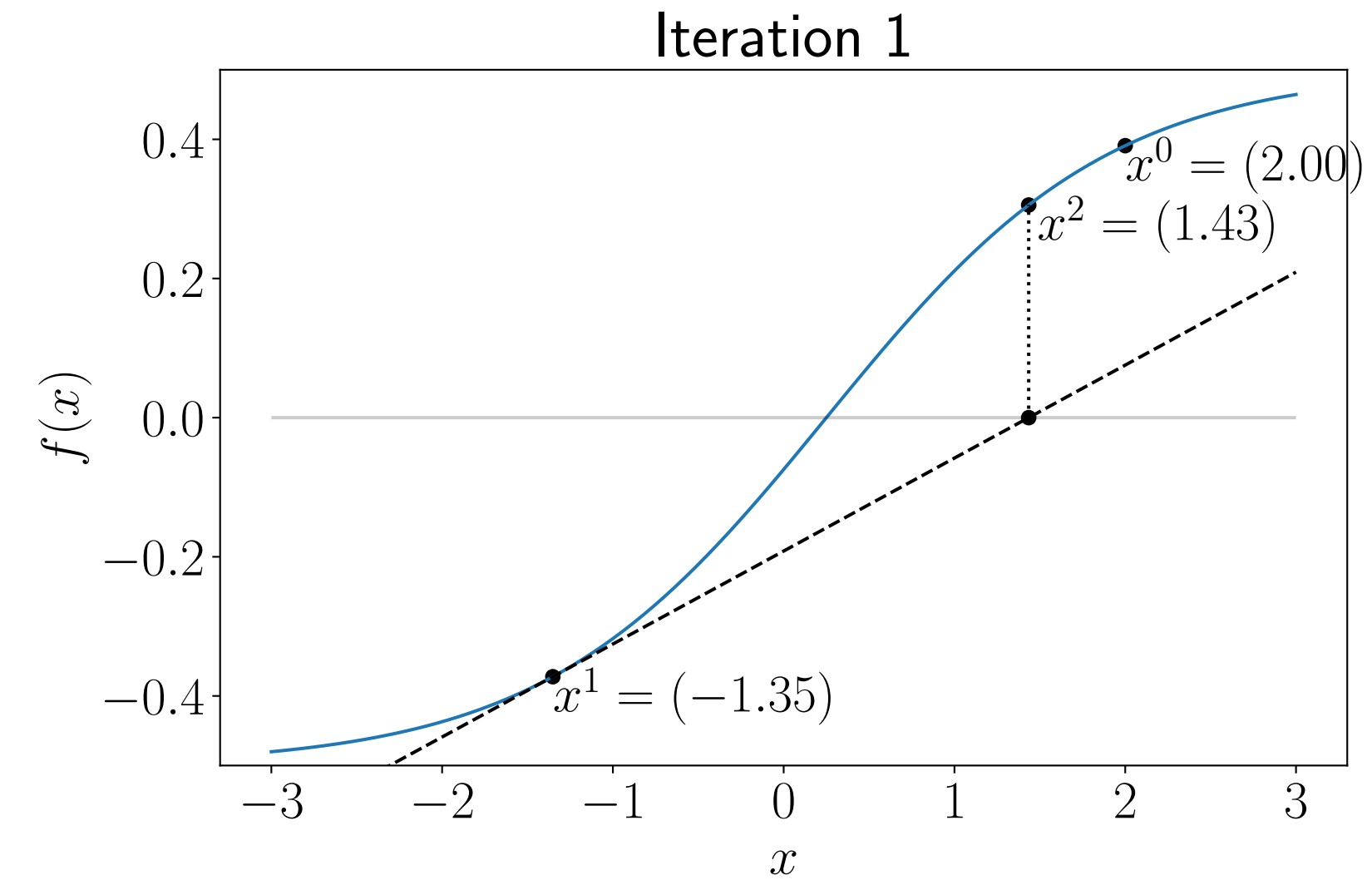
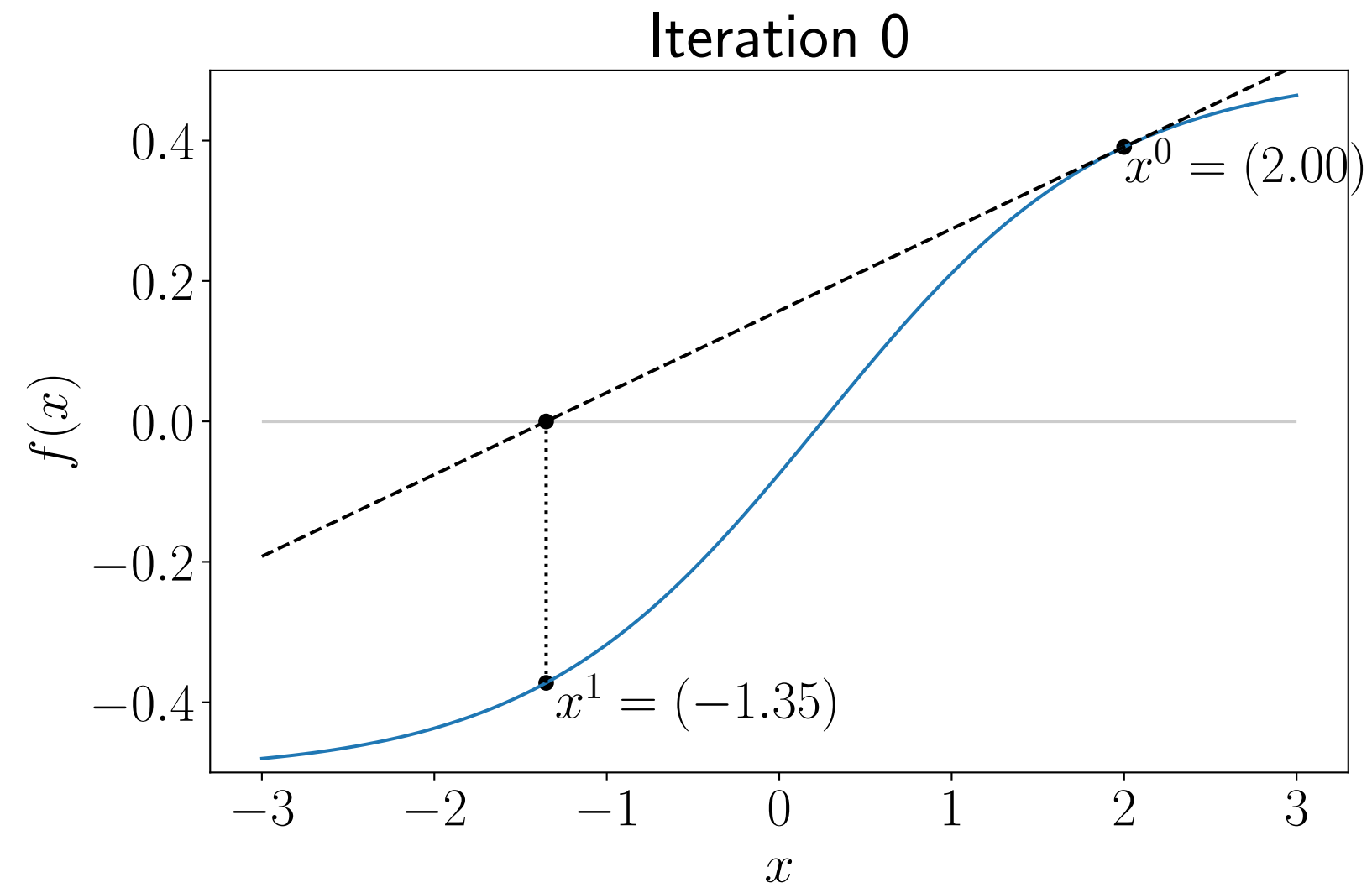
# Newton's method example

$$f(x) = \frac{1}{1 + e^{-1.2x+0.3}} - 0.5$$

$$f(x) = 0$$



$$x^* = 0.3$$



# Newton's root finding method (multivariable)

**Goal:** solve

$$h(x) = 0$$

# Newton's root finding method (multivariable)

**Goal:** solve  
 $h(x) = 0$

$$h: \mathbb{R}^n \rightarrow \mathbb{R}^m$$

## Method

1. Make a guess  $x^k$  and a linear approximation

$$h(x) \approx \underbrace{h(x^k) + Dh(x^k)(x - x^k)}_{\hat{h}(x)}$$

2. Iteratively set  $\hat{h}(x)$  to 0

$$h(x^k) + Dh(x^k)(x^{k+1} - x^k) = 0$$

## Derivative

$$Dh = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \vdots & \vdots \\ \frac{\partial h_m}{\partial x_1} & \cdots & \frac{\partial h_m}{\partial x_n} \end{bmatrix}$$

# Newton method iterations

$$h(x^k) + Dh(x^k)(x^{k+1} - x^k) = 0$$

# Newton method iterations

$$h(x^k) + Dh(x^k) \underbrace{(x^{k+1} - x^k)}_{\Delta x} = 0$$

# Newton method iterations

$$x^k + \Delta x = \cancel{x^k} + x^{k+1} - \cancel{x^k} = x^{k+1}$$

$$h(x^k) + Dh(x^k) \underbrace{(x^{k+1} - x^k)}_{\Delta x} = 0$$

## Iterations

- Solve  $Dh(x^k) \Delta x = -h(x^k)$
- $x^{k+1} \leftarrow x^k + \Delta x$

# Newton method iterations

$$h(x^k) + Dh(x^k) \underbrace{(x^{k+1} - x^k)}_{\Delta x} = 0$$

## Iterations

- Solve  $Dh(x^k)\Delta x = -h(x^k)$
- $x^{k+1} \leftarrow x^k + \Delta x$

## Remarks

- Iterations can be **expensive** (linear system solution)
- **Fast convergence** close to the solution  $x^*$



# Linear optimization as a root finding problem

## Optimality conditions

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \end{array}$$

# Linear optimization as a root finding problem

## Optimality conditions

|                        | <b>Primal</b>                         | <b>Dual</b>                              |
|------------------------|---------------------------------------|--|
| minimize $c^T x$       | minimize $c^T x$                      | maximize $-b^T y$                        |
| subject to $Ax \leq b$ | subject to $Ax + s = b$<br>$s \geq 0$ | subject to $A^T y + c = 0$<br>$y \geq 0$ |

# Linear optimization as a root finding problem

## Optimality conditions

|            | Primal      | Dual       |                 |
|------------|-------------|------------|-----------------|
| minimize   | $c^T x$     | maximize   | $-b^T y$        |
| subject to | $Ax \leq b$ | subject to | $A^T y + c = 0$ |
|            | $s \geq 0$  |            | $y \geq 0$      |

$s = b - Ax$

### KKT conditions

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$$s_i y_i = 0, \quad i = 1, \dots, m$$

$$s, y \geq 0$$

$y_i (b - Ax)_i = 0$

# Linear optimization as a root finding problem

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$$s_i y_i = 0, \quad i = 1, \dots, m$$

$$s, y \geq 0$$

# Linear optimization as a root finding problem

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$$s_i y_i = 0, \quad i = 1, \dots, m$$

$$s, y \geq 0$$

## Diagonalize complementary slackness

$$S = \text{diag}(s) = \begin{bmatrix} s_1 & & & \\ & s_2 & & \\ & & \ddots & \\ & & & s_m \end{bmatrix}$$
$$Y = \text{diag}(y) = \begin{bmatrix} y_1 & & & \\ & y_2 & & \\ & & \ddots & \\ & & & y_m \end{bmatrix}$$

$$SY\mathbf{1} = \text{diag}(s)\text{diag}(y)\mathbf{1} = \begin{bmatrix} s_1 y_1 & & & \\ & s_2 y_2 & & \\ & & \ddots & \\ & & & s_m y_m \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} s_1 y_1 \\ s_2 y_2 \\ \vdots \\ s_m y_m \end{bmatrix}$$

# Linear optimization as a root finding problem

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$$s_i y_i = 0, \quad i = 1, \dots, m$$

$$s, y \geq 0$$

## Diagonalize complementary slackness

$$S = \text{diag}(s) = \begin{bmatrix} s_1 & & & \\ & s_2 & & \\ & & \ddots & \\ & & & s_m \end{bmatrix}$$

$$Y = \text{diag}(y) = \begin{bmatrix} y_1 & & & \\ & y_2 & & \\ & & \ddots & \\ & & & y_m \end{bmatrix}$$

$$SY\mathbf{1} = \text{diag}(s)\text{diag}(y)\mathbf{1} = \begin{bmatrix} s_1 y_1 & & & \\ & s_2 y_2 & & \\ & & \ddots & \\ & & & s_m y_m \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} s_1 y_1 \\ s_2 y_2 \\ \vdots \\ s_m y_m \end{bmatrix}$$

$$s_i y_i = 0, \quad i = 1, \dots, m \quad \iff \quad SY\mathbf{1} = 0$$

# Main idea

## Optimality conditions

$$h(y, x, s) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} \end{bmatrix} = \begin{bmatrix} r_p \\ r_d \\ SY\mathbf{1} \end{bmatrix} = 0 \quad \begin{array}{l} S = \mathbf{diag}(s) \\ Y = \mathbf{diag}(y) \end{array}$$

$s, y \geq 0$

- Apply variants of Newton's method to solve  $h(x, s, y) = 0$
- Enforce  $s, y > 0$  (strictly) at every iteration
- **Motivation** avoid getting stuck in “corners”

# Newton's method for optimality conditions

## Optimality conditions

$$h(y, x, s) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} \end{bmatrix} = \begin{bmatrix} r_p \\ r_d \\ SY\mathbf{1} \end{bmatrix} = 0$$
$$s, y \geq 0$$



# Newton's method for optimality conditions

Optimality conditions

$$h(y, x, s) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} \end{bmatrix} = \begin{bmatrix} r_p \\ r_d \\ SY\mathbf{1} \end{bmatrix} = 0$$

$s, y \geq 0$

Derivative

$$Dh(y, x, s) = \begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix}$$

# Newton's method for optimality conditions

## Optimality conditions

$$h(y, x, s) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} \end{bmatrix} = \begin{bmatrix} r_p \\ r_d \\ SY\mathbf{1} \end{bmatrix} = 0$$

$s, y \geq 0$

## Derivative

$$Dh(y, x, s) = \begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix}$$

## Iterations

- Solve  $Dh(y^k, x^k, s^k) \Delta(y^k, x^k, s^k) = -h(y^k, x^k, s^k)$

- $\begin{bmatrix} y^{k+1} \\ x^{k+1} \\ s^{k+1} \end{bmatrix} \leftarrow \begin{bmatrix} y^k \\ x^k \\ s^k \end{bmatrix} + \Delta(y^k, x^k, s^k)$

# Newton's method for optimality conditions

## Optimality conditions

$$h(y, x, s) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} \end{bmatrix} = \begin{bmatrix} r_p \\ r_d \\ SY\mathbf{1} \end{bmatrix} = 0$$

$$s, y \geq 0$$

## Derivative

$$Dh(y, x, s) = \begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix}$$

## Iterations

- Solve  $Dh(y^k, x^k, s^k)\Delta(y^k, x^k, s^k) = -h(y^k, x^k, s^k)$

- $\begin{bmatrix} y^{k+1} \\ x^{k+1} \\ s^{k+1} \end{bmatrix} \leftarrow \begin{bmatrix} y^k \\ x^k \\ s^k \end{bmatrix} + \Delta(y^k, x^k, s^k)$

**Caution!**

It might make  $(s, y)$  negative!

**Central path**

# Line search to stay feasible

## Root-finding equation

$$h(y, x, s) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} \end{bmatrix} = \begin{bmatrix} r_p \\ r_d \\ SY\mathbf{1} \end{bmatrix} = 0$$

## Linear system

$$\begin{matrix} Dh \\ \begin{bmatrix} 0 \\ A^T \\ S \end{bmatrix} \end{matrix} \begin{bmatrix} A & I & 0 \\ 0 & 0 & 0 \\ 0 & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -h \\ -r_p \\ -r_d \\ -SY\mathbf{1} \end{bmatrix}$$

## Residuals

$$r_p = Ax + s - b$$

$$r_d = A^T y + c$$

# Line search to stay feasible

## Root-finding equation

$$h(y, x, s) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} \end{bmatrix} = \begin{bmatrix} r_p \\ r_d \\ SY\mathbf{1} \end{bmatrix} = 0$$

## Linear system

$$\begin{matrix} Dh & & & -h \\ \begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} & \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} & = & \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} \end{bmatrix} \end{matrix}$$

## Residuals

$$r_p = Ax + s - b$$

$$r_d = A^T y + c$$

**Line search to enforce  $s, y > 0$**

$$(y, x, s) \leftarrow (y, x, s) + \alpha(\Delta y, \Delta x, \Delta s)$$

# Line search to stay feasible

## Root-finding equation

$$h(y, x, s) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} \end{bmatrix} = \begin{bmatrix} r_p \\ r_d \\ SY\mathbf{1} \end{bmatrix} = 0$$

## Linear system

$$\begin{matrix} Dh \\ \begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \end{matrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{matrix} -h \\ \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} \end{bmatrix} \end{matrix}$$

## Residuals

$$r_p = Ax + s - b$$

$$r_d = A^T y + c$$

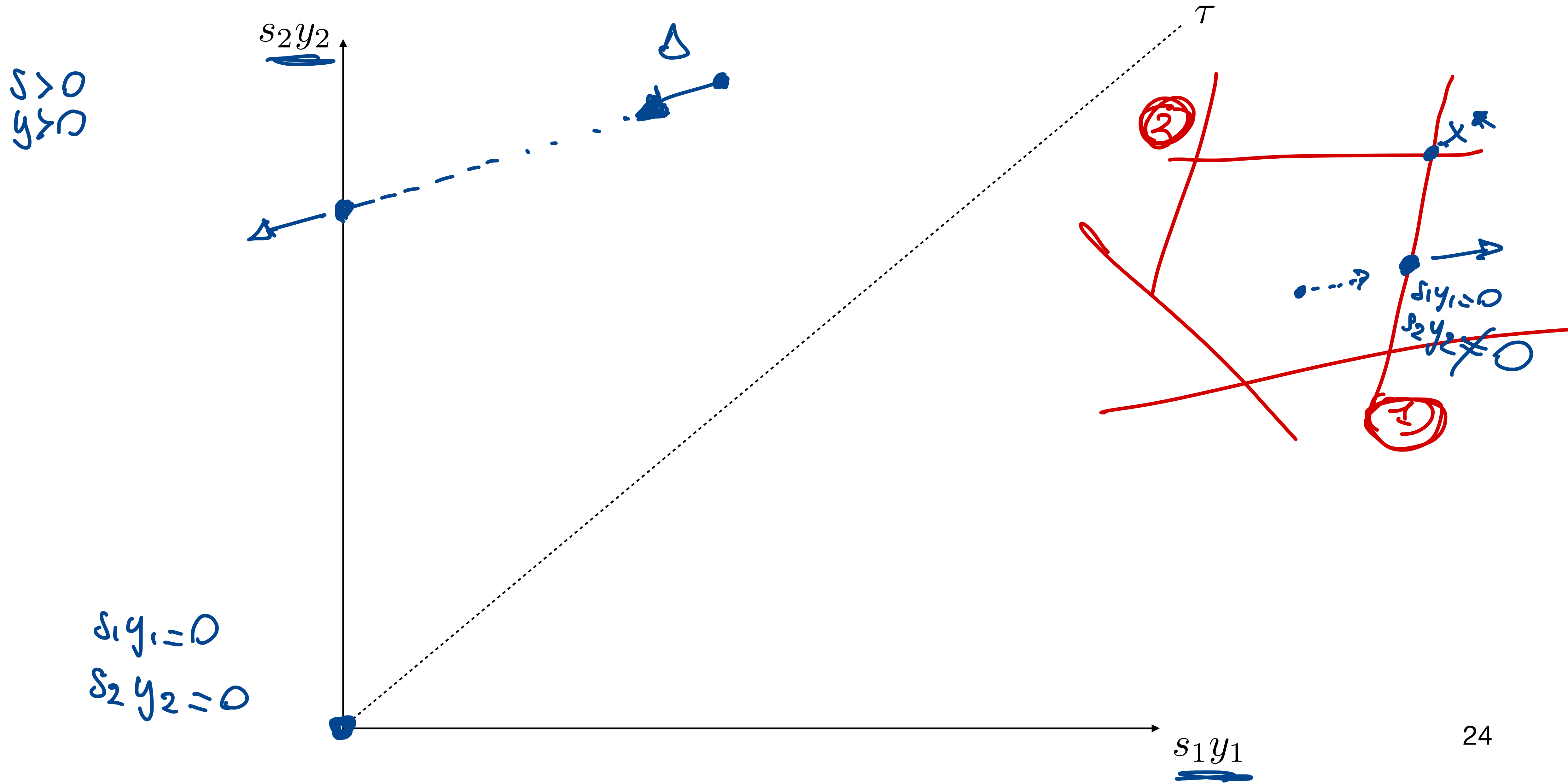
## Issue

Pure **Newton's step** does not allow significant progress towards

$$h(y, x, s) = 0 \text{ and } s, y \geq 0.$$

**Line search** to enforce  $s, y > 0$   
 $(y, x, s) \leftarrow (y, x, s) + \alpha(\Delta y, \Delta x, \Delta s)$

# The central path





# Smoothed optimality conditions

## Optimality conditions

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$$s_i y_i = \tau \longleftarrow \text{Same } \tau \text{ for every pair}$$

$$s, y \geq 0$$

Same optimality conditions for a “smoothed” version of our problem

# Smoothed optimality conditions

## Optimality conditions

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$$s_i y_i = \tau \longleftarrow \text{Same } \tau \text{ for every pair}$$

$$s, y \geq 0$$

Same optimality conditions for a “smoothed” version of our problem

## Duality gap

$$m\tau = s^T y = (b - Ax)^T y = b^T x - x^T A^T y = b^T y + c^T x$$

# Newton's method for smoothed optimality conditions

## Smoothed optimality conditions

$$h_{\tau}(y, x, s) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} - \tau\mathbf{1} \end{bmatrix} = 0$$
$$s, y \geq 0$$

# Newton's method for smoothed optimality conditions

## Smoothed optimality conditions

$$h_{\tau}(y, x, s) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} - \tau\mathbf{1} \end{bmatrix} = 0$$

$$s, y \geq 0$$

## Linear system

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY + \tau\mathbf{1} \end{bmatrix}$$

**Line search** to enforce  $s, y > 0$

$$(y, x, s) \leftarrow (y, x, s) + \alpha(\Delta y, \Delta x, \Delta s)$$

# The path parameter

## Duality measure

$$\mu = \frac{s^T y}{m} \quad (\text{average value of the pairs } s_i y_i)$$

# The path parameter

## Duality measure

$$\mu = \frac{s^T y}{m} \quad (\text{average value of the pairs } s_i y_i)$$

## Linear system

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} + \sigma\mu\mathbf{1} \end{bmatrix}$$

# The path parameter

## Duality measure

$$\mu = \frac{s^T y}{m} \quad (\text{average value of the pairs } s_i y_i)$$

## Linear system

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} + \sigma\mu\mathbf{1} \end{bmatrix}$$

## Centering parameter

$$\sigma \in [0, 1]$$

# The path parameter

## Duality measure

$$\mu = \frac{s^T y}{m} \quad (\text{average value of the pairs } s_i y_i)$$

## Linear system

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} + \sigma\mu\mathbf{1} \end{bmatrix}$$

## Centering parameter

$$\sigma \in [0, 1]$$

$$\sigma = 0 \Rightarrow$$

Newton step

$$\sigma = 1 \Rightarrow$$

Centering step towards  $(y^*(\mu), x^*(\mu), s^*(\mu))$

$$\boxed{\nu = \mu}$$



# The path parameter

## Duality measure

$$\mu = \frac{s^T y}{m} \quad (\text{average value of the pairs } s_i y_i)$$

## Linear system

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} + \sigma\mu\mathbf{1} \end{bmatrix}$$

## Centering parameter

$$\sigma \in [0, 1]$$

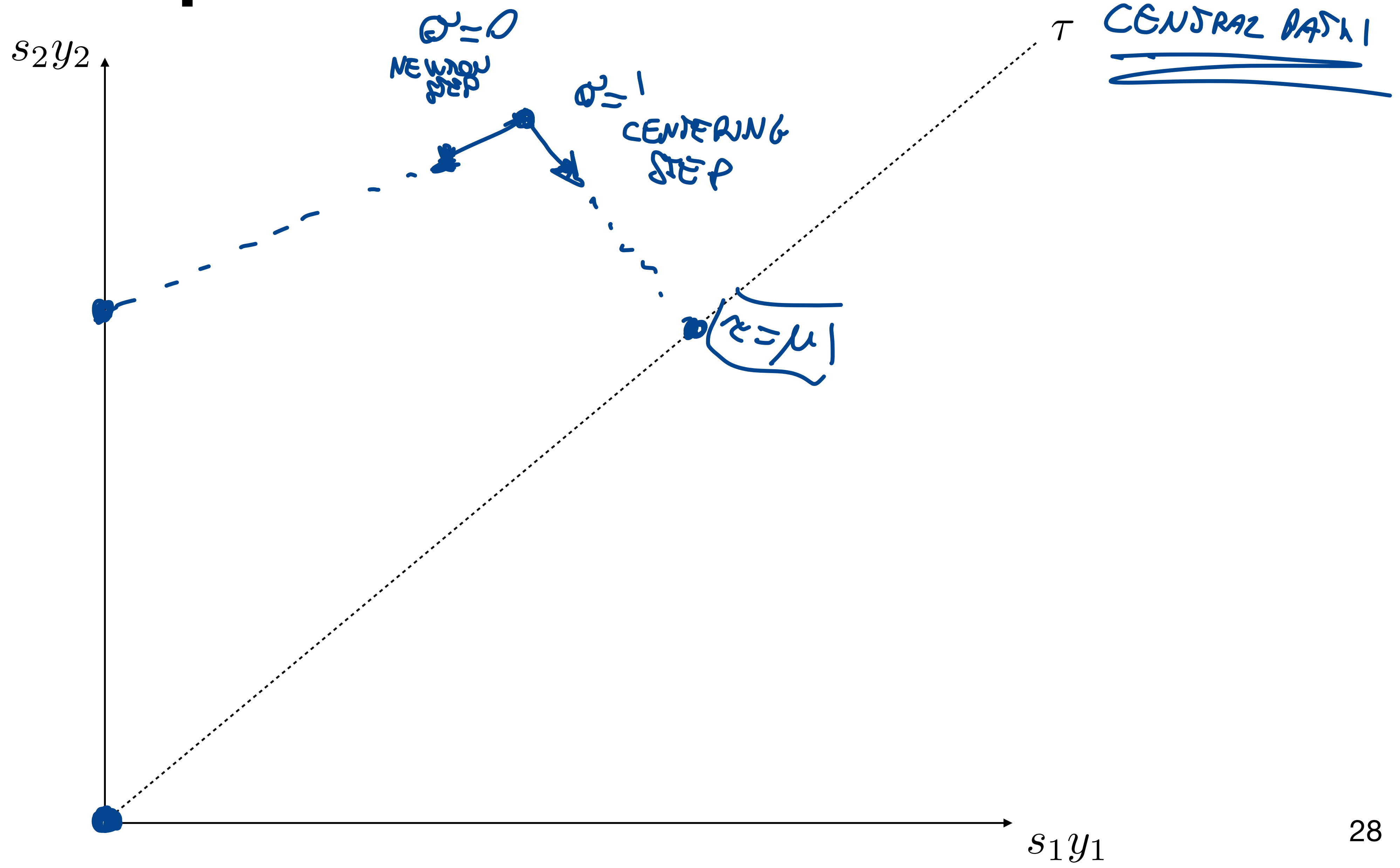
$\sigma = 0 \Rightarrow$  Newton step

$\sigma = 1 \Rightarrow$  Centering step towards  $(y^*(\mu), x^*(\mu), s^*(\mu))$

**Line search** to enforce  $s, y > 0$

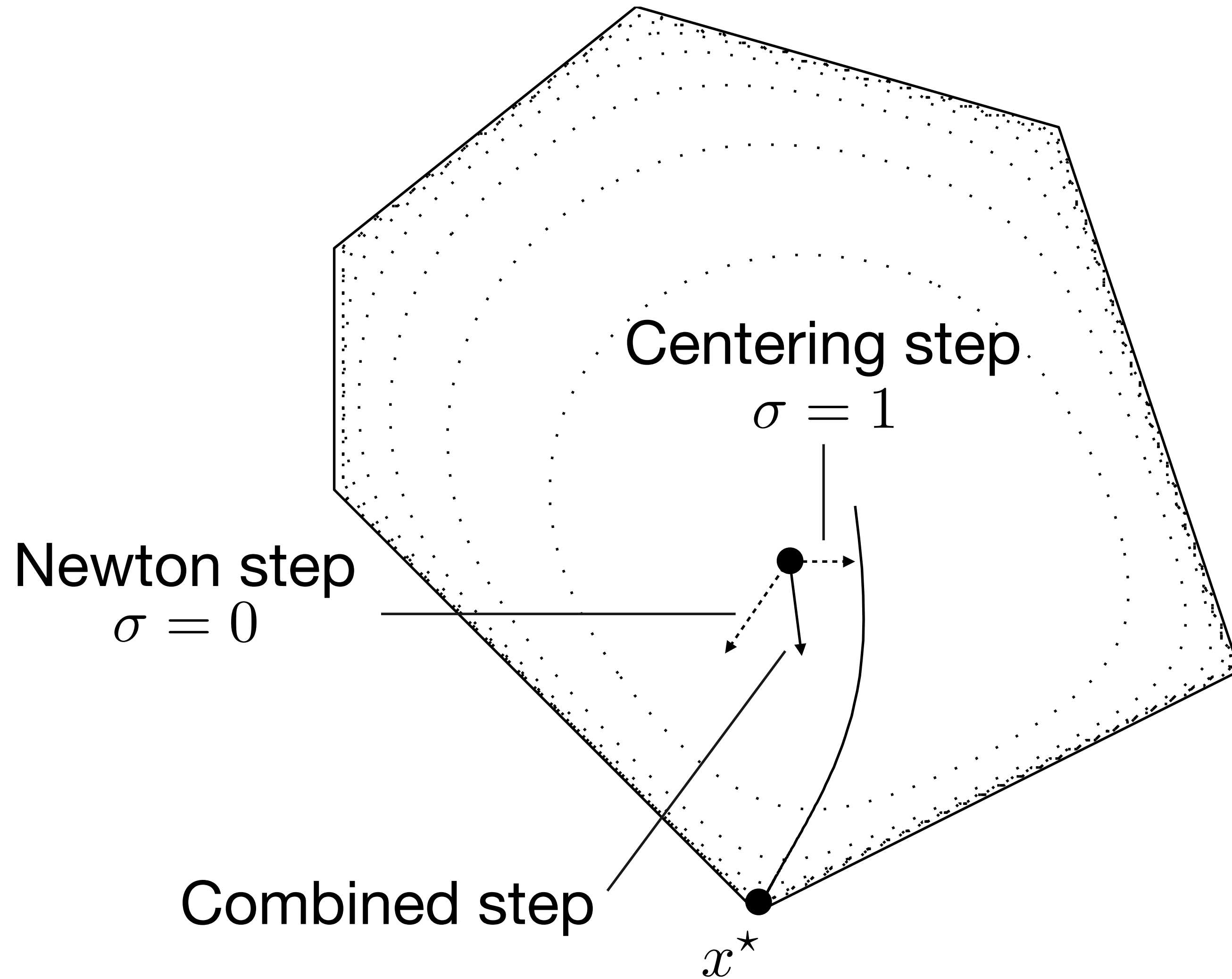
$$(y, x, s) \leftarrow (y, x, s) + \alpha(\Delta y, \Delta x, \Delta s)$$

# The central path

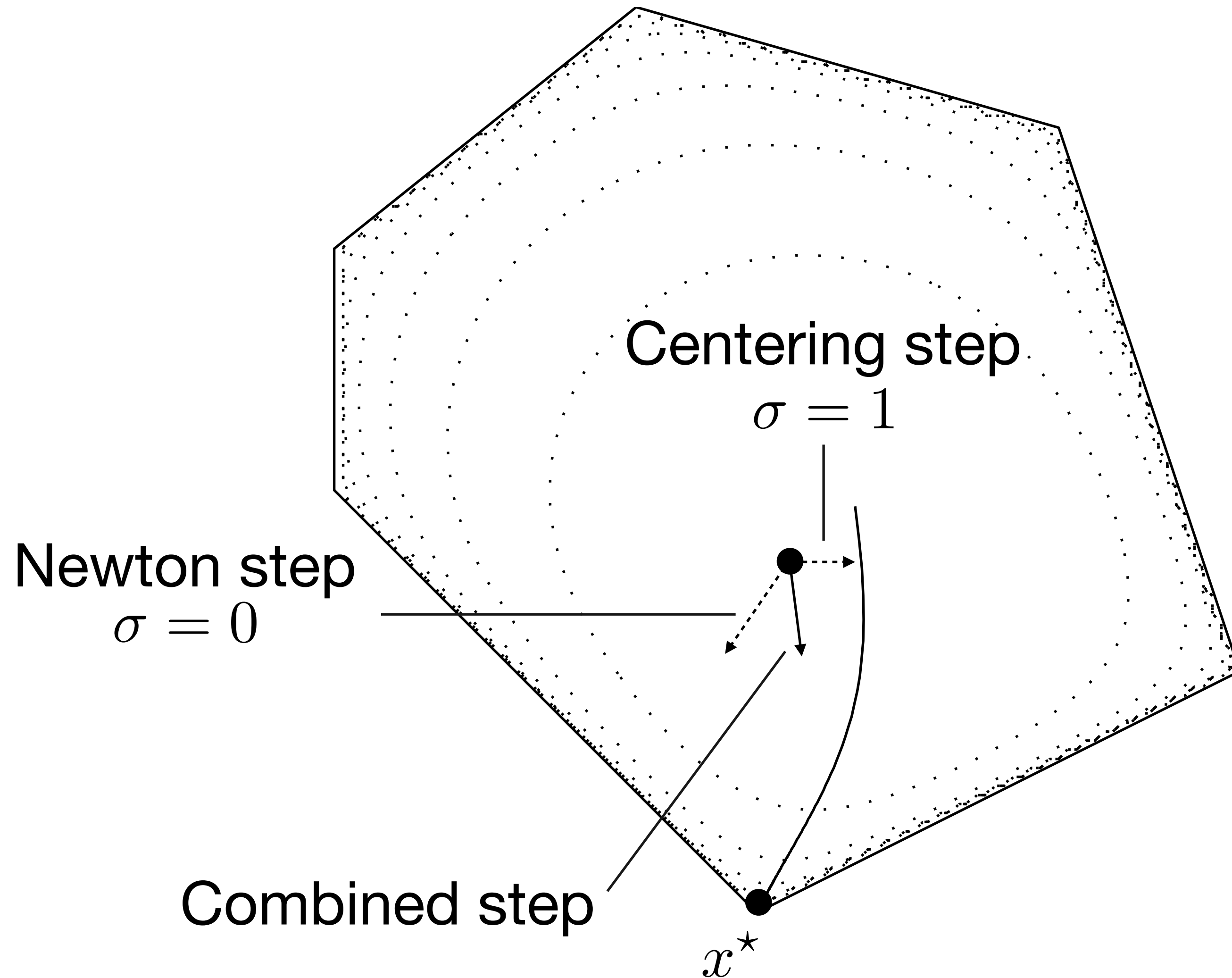


# Primal-dual path-following method

# Path-following algorithm idea



# Path-following algorithm idea

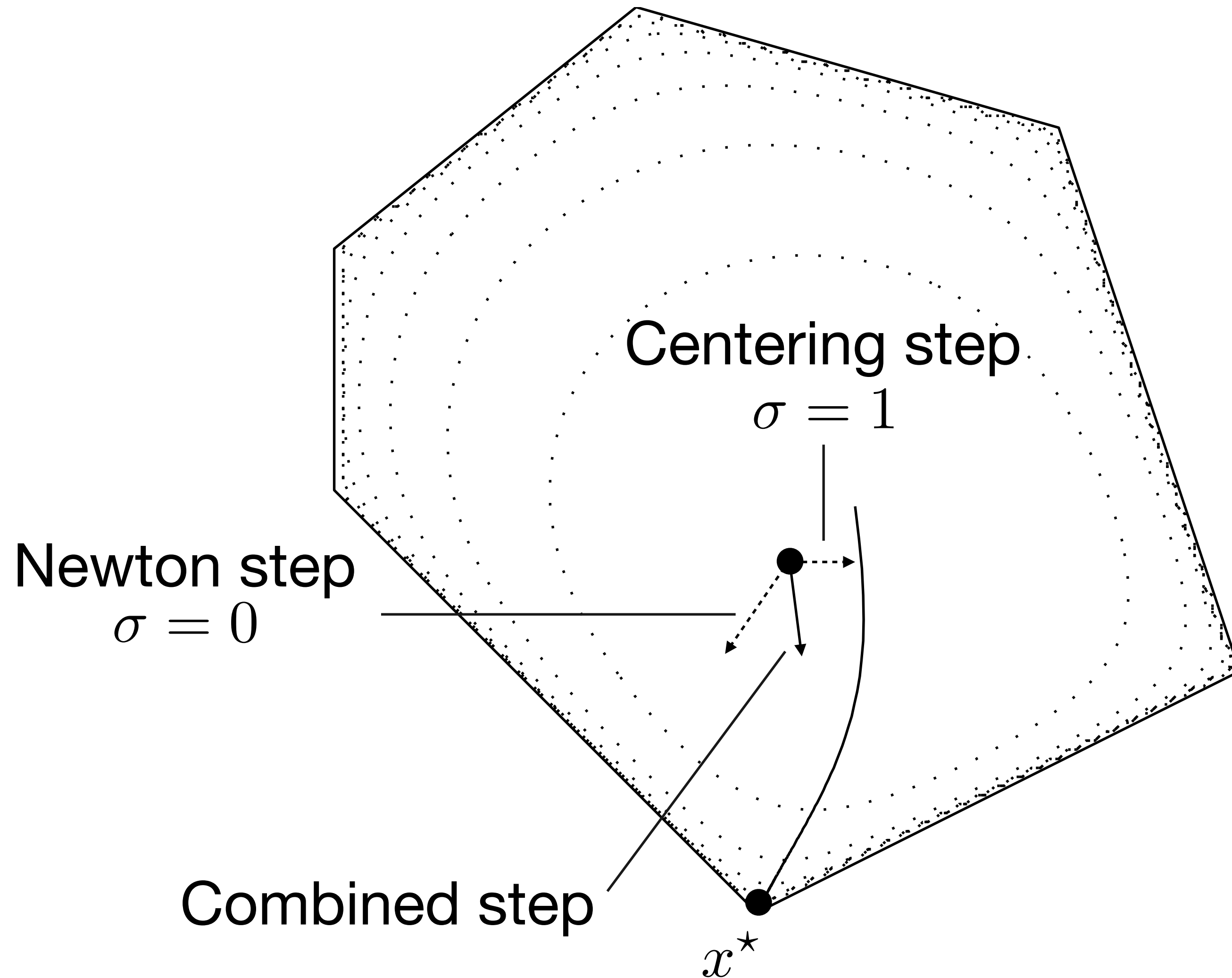


## Centering step

It brings towards the **central path** and is usually biased towards  $s, y > 0$ .

**No progress** on duality measure  $\mu$

# Path-following algorithm idea



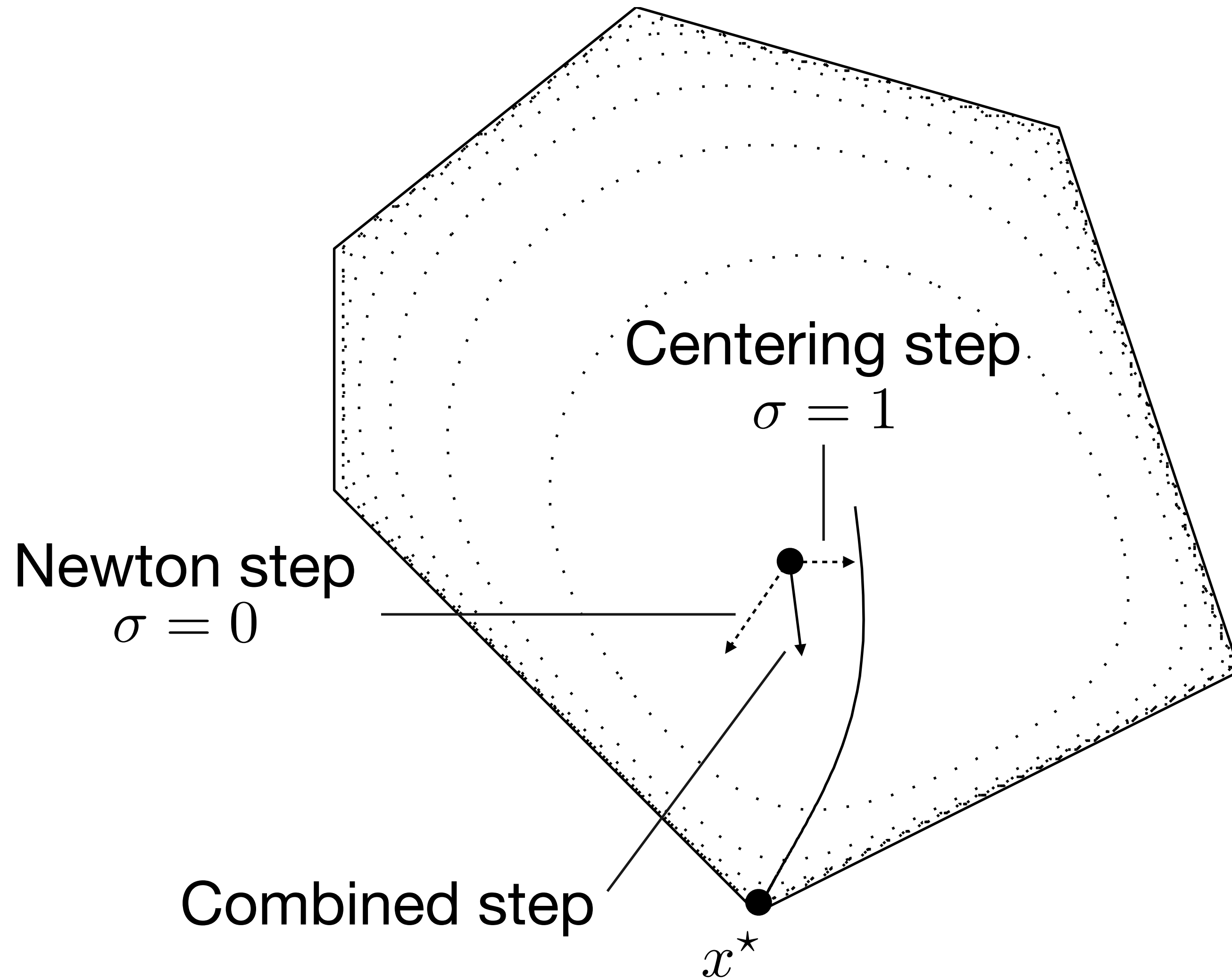
## Centering step

It brings towards the **central path** and is usually biased towards  $s, y > 0$ .  
**No progress** on duality measure  $\mu$

## Newton step

It brings towards the **zero duality measure**  $\mu$ . Quickly violates  $s, y > 0$ .

# Path-following algorithm idea



## Centering step

It brings towards the **central path** and is usually biased towards  $s, y > 0$ .  
**No progress** on duality measure  $\mu$

## Newton step

It brings towards the **zero duality measure**  $\mu$ . Quickly violates  $s, y > 0$ .

## Combined step

Best of both worlds with longer steps

# Primal-dual path-following algorithm

## Initialization

1. Given  $(x_0, s_0, y_0)$  such that  $s_0, y_0 > 0$

## Iterations

1. Choose  $\sigma \in [0, 1]$

2. Solve 
$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} + \sigma\mu\mathbf{1} \end{bmatrix}$$
 where  $\mu = s^T y / m$

3. Find maximum  $\alpha$  such that  $y + \alpha\Delta y > 0$  and  $s + \alpha\Delta s > 0$

4. Update  $(y, x, s) \leftarrow (y, x, s) + \alpha(\Delta y, \Delta x, \Delta s)$



# Working towards optimality conditions

Optimality conditions satisfied **only at convergence**

## Primal residual

$$r_p = Ax + s - b \rightarrow 0$$

## Dual residual

$$r_d = A^T y + c \rightarrow 0$$

## Complementary slackness

$$s^T y \rightarrow 0$$

# Working towards optimality conditions

Optimality conditions satisfied **only at convergence**

## Primal residual

$$r_p = Ax + s - b \rightarrow 0$$

## Dual residual

$$r_d = A^T y + c \rightarrow 0$$

## Complementary slackness

$$s^T y \rightarrow 0$$

## Stopping criteria

$$\|r_p\| \leq \epsilon_{\text{pri}}$$

$$\|r_d\| \leq \epsilon_{\text{dua}}$$

$$s^T y \leq \epsilon_{\text{gap}}$$

# Logarithmic barrier functions

# Smoothed optimality conditions

## Optimality conditions

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$$s_i y_i = \tau \longleftarrow \text{Same } \tau \text{ for every pair}$$

$$s, y \geq 0$$

Same optimality conditions for a “smoothed” version of our problem

# Smoothed optimality conditions

## Optimality conditions

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$$s_i y_i = \tau \longleftarrow \text{Same } \tau \text{ for every pair}$$

$$s, y \geq 0$$

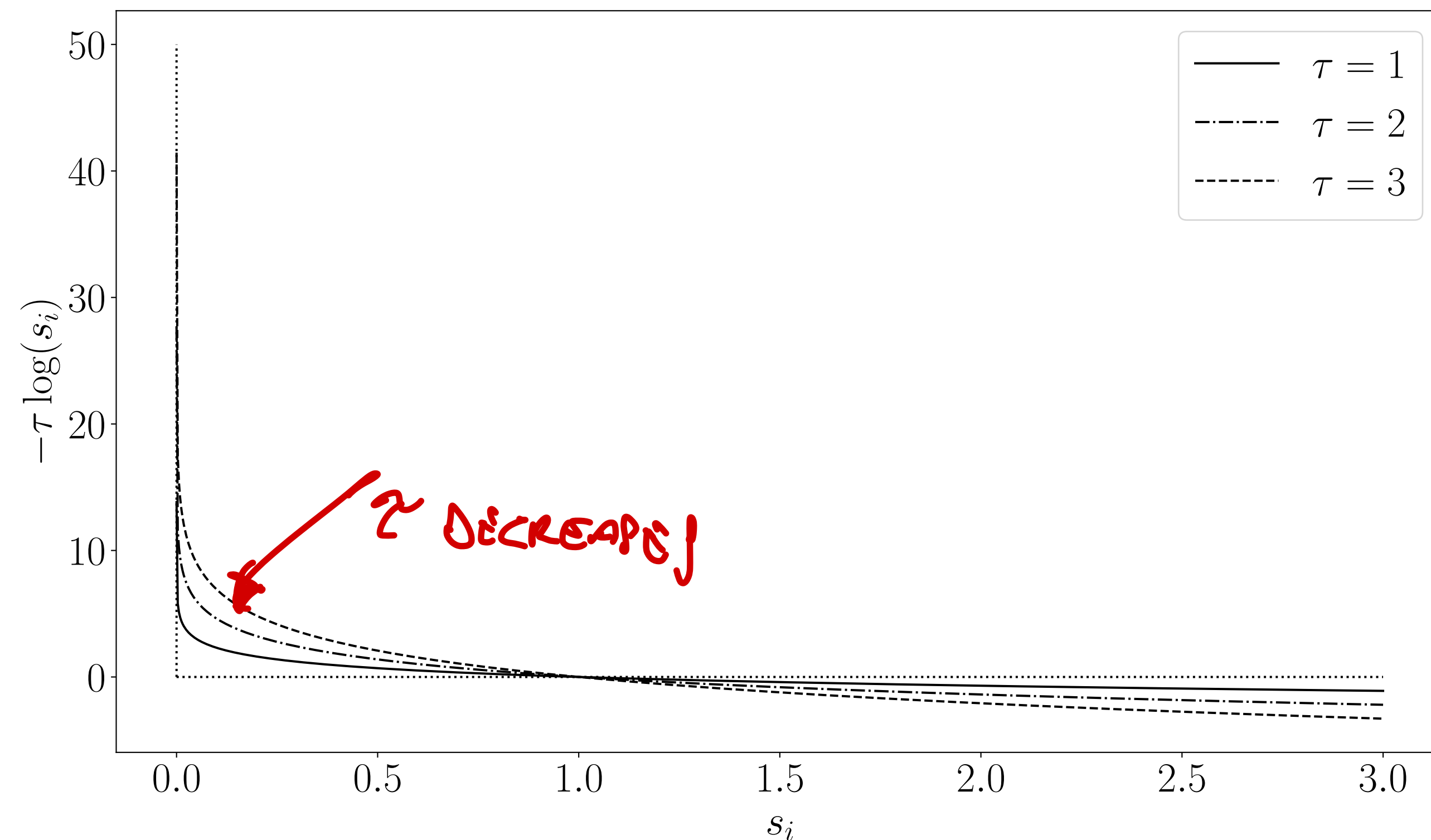
Same optimality conditions for a “smoothed” version of our problem

**Do solutions actually exist?**

**What do they represent?**

# Logarithmic barrier

$$\phi(s) = -\tau \sum_{i=1}^m \log(s_i) \quad \text{on domain} \quad s_i > 0$$



As  $\tau \rightarrow 0$  it approximates

$$\mathcal{I}_{s_i \geq 0} = \begin{cases} 0 & \text{if } s_i \geq 0 \\ \infty & \text{otherwise} \end{cases}$$

# Smoothed problem

minimize  $c^T x$

subject to  $Ax + s = b$

$s \geq 0$

# Smoothed problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax + s = b \\ & s \geq 0 \end{array} \longrightarrow \begin{array}{ll} \text{minimize} & c^T x + \phi(s) = c^T x - \tau \sum_{i=1}^m \log(s_i) \\ \text{subject to} & Ax + s = b \end{array}$$



# Smoothed problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax + s = b \\ & s \geq 0 \end{array} \longrightarrow \begin{array}{ll} \text{minimize} & c^T x + \phi(s) = c^T x - \tau \sum_{i=1}^m \log(s_i) \\ \text{subject to} & Ax + s = b \end{array}$$

## Lagrangian function

$$L(x, s, y) = c^T x - \tau \sum_{i=1}^m \log(s_i) + y^T (Ax + s - b)$$

# Smoothed problem

$$\begin{array}{l} \text{minimize} \quad c^T x \\ \text{subject to} \quad Ax + s = b \\ \quad \quad \quad s \geq 0 \end{array} \longrightarrow \begin{array}{l} \text{minimize} \quad c^T x + \phi(s) = c^T x - \tau \sum_{i=1}^m \log(s_i) \\ \text{subject to} \quad Ax + s = b \end{array}$$

## Lagrangian function

$$L(x, s, y) = c^T x - \tau \sum_{i=1}^m \log(s_i) + y^T (Ax + s - b)$$

$$\frac{\partial L}{\partial x} = A^T y + c = 0$$

$$\frac{\partial L}{\partial s_i} = -\tau \frac{1}{s_i} + y_i = 0 \quad \implies s_i y_i = \tau$$

# Central path

$$\begin{aligned} \text{minimize} \quad & c^T x - \tau \sum_{i=1}^m \log(s_i) \\ \text{subject to} \quad & Ax + s = b \end{aligned}$$

Set of points  $(x^*(\tau), s^*(\tau), y^*(\tau))$   
with  $\tau > 0$  such that

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$$s_i y_i = \tau$$

$$s, y \geq 0$$

# Central path

$$\begin{aligned} &\text{minimize} && c^T x - \tau \sum_{i=1}^m \log(s_i) \\ &\text{subject to} && Ax + s = b \end{aligned}$$

Set of points  $(x^*(\tau), s^*(\tau), y^*(\tau))$   
with  $\tau > 0$  such that

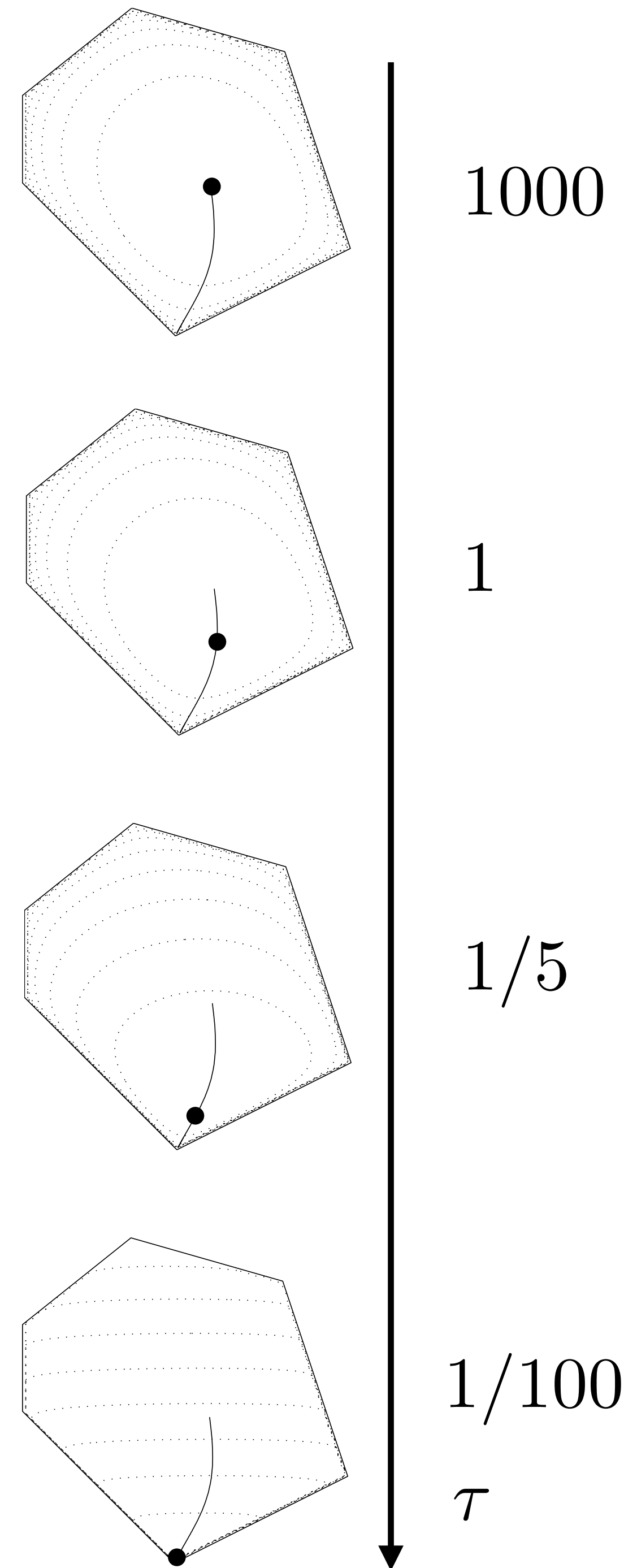
$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$$s_i y_i = \tau$$

$$s, y \geq 0$$

**Analytic  
Center**  
 $\tau \rightarrow \infty$



**Main idea**

Follow central path as  $\tau \rightarrow 0$

# Interior-point methods for linear optimization

Today, we learned to:

- **Apply** Newton's method to solve optimality conditions
- **Follow** the central path and the smoothed optimality conditions
- **Use logarithmic barrier functions** to interpret central path steps

# References

- D. Bertsimas and J. Tsitsiklis: Introduction to Linear Optimization
  - Chapter 9.4 — 9.6: Interior point methods
- R. Vanderbei: Linear Programming
  - Chapter 17: The Central Path
  - Chapter 15: A Path-Following Method

# Next lecture

- Practical interior-point method (Mehrotra predictor-corrector algorithm)
- Implementation details
- Interior-point vs simples