

cykel finns, betraktar vi optimallösningen som obegränsad (åk runt i cykeln oändligt många gånger, vilket ger en oändlig vinst). Detta motsvarar en problemformulering med nodjämviktssvillkor och ickenegativitetsvillkor.

Man kan dock tycka att denna problemformulering är otillfredsställande, om man säkert vill ha en väg från start- till slutnoden, även om en negativ cykel existerar. I så fall söker vi den billigaste enkla vägen från start- till slutnod.

Igenkänningsversionen av billigaste enkla väg-problemet är följande: Finns en enkel väg från start- till slutnod med $c^T x \leq L$? Detta problem är faktiskt *NP*-fullständigt (om negativa cykler kan finnas). En viss förståelse för detta faktum kan fås om man inser att de subtursförbjudande bivillkor man kan använda i handelsresandeproblemet skulle behöva läggas till i billigaste enkla väg-problemet också.

Ett annat problem där komplexiteten beror på bågkostnaderna är det så kallade *billigaste cykel-problemet*, (på engelska *minimal weighted girth problem*, vilket kanske borde översättas med "billigaste gördel-problemet"). Uppgiften är att finna den billigaste enkla cykeln i en oriktad graf med bågkostnader. Till skillnad mot handelsresandeproblemet måste inga noder passeras. Dock brukar man kräva att cykeln ska innehålla minst tre noder (för att få kallas cykel). Om bågkostnaderna är icke-negativa, blir problemet lösbart i polynomisk tid. Detta gäller även för fallet där den totala kostnaden för varje cykel i grafen är ickenegativ (vilket dock inte alltid är så lätt att kontrollera).

Men utan dessa begränsningar är problemet *NP*-svårt. Detta kan man inse genom att utgå från ett oriktat handelsresandeproblem och dra bort en stor negativ konstant från varje bågkostnad. (Som bekant ändrar detta inte optimallösningen.) Genom att lösa billigaste cykel-problemet kan man på detta sätt lösa handelsresandeproblemet.

15.2 Övningsuppgifter

Några övningsuppgifter angående komplexitet ges nedan.

UPPGIFT 15.1

För varje deluppgift nedan, ange om problemet tillhör *P*, om problemet tillhör *NP* och om problemet är *NP*-fullständigt. Ge korta, kärnfulla motiveringar, gärna med kända probleminamn, inte fullständiga bevis. (Samtliga koefficienter nedan antas vara positiva och heltaliga. För grafer betecknar n antalet noder i grafen.)

a) Vi har k st olika sandsäckar med vikterna v_j för $j = 1, \dots, k$. Kan man packa ner alla sandsäckarna i p stycken lådor, utan att någon låda får en totalvikt större än q ? (q är ett positivt heltal, $q > v_j$ och $pq > \sum_j v_j$.)

b) Samma fråga som i uppgift a, men vi får öppna sandsäckarna och hälla i sanden i lådorna. (Sanden från en säck får hamna i fler än en låda.)

- c) Finns det en delmängd av (hela) sandsäckar i uppgift a som tillsammans har exakt vikten q ? (q är heltal och $0 < q < \sum_j v_j$.)
- d) Finns det en väg som börjar i nod s och slutar i nod t och innehåller varje nod exakt en gång i en given oriktad graf?
- e) Finns det en väg som börjar och slutar i nod s och innehåller varje båge exakt en gång i en given oriktad graf?
- f) Finns det en väg som börjar i nod s och slutar i nod t och inte innehåller fler än q bågar i en given riktad graf? ($3 < q < n$.)
- g) Finns det en enkel cykel (dvs. där ingen nod har valens större än två) som innehåller minst q bågar i en given oriktad graf? ($3 < q < n$.)
- h) Finns det ett sätt att skicka q enheter flöde från nod s till nod t i en given riktad graf med bågkapaciteter u_{ij} ? ($0 < u_{ij} < q$.)
- i) Finns det ett uppspännande träd där ingen nodvalens överstiger q i en given oriktad graf? ($2 < q < n$.)
- j) Kan man välja ut q st noder i en given oriktad graf så att ingen vald nod har en direktbåge till någon annan vald nod? ($3 < q < n$.)
- k) Kan man välja ut q st noder i en given oriktad graf så att samtliga bågar är anslutna till minst en vald nod? ($3 < q < n$.)
- l) Kan man välja ut q st noder i en given oriktad graf så att alla noder i grafen är uppnåeliga från någon av de valda noderna? ($3 < q < n$.)

UPPGIFT 15.2

I en viss liten elektrisk apparat finns följande nätverk. I ena änden finns ett antal ingångar, där man kan välja att lägga spänning (500 volt) eller inte (0 volt), och i andra änden finns en utgång. Bågarna i nätverket är helt enkelt elektriska ledare, och målet är att få spänning i utgången. Noderna består av tre olika typer av aktiva komponenter. En V-nod ("vänd") har en ingång och en utgång, och utsignalen är helt enkelt insignalen omvänd (spänning byts mot ingen spänning och vice versa). En N-nod ("någon") har flera ingångar och en utgång, och utgången får spänning om någon av ingångarna har spänning. En A-nod ("alla") har också den flera ingångar och en utgång, och utgången får spänning om alla ingångarna har spänning. Givet en viss konfiguration (noder, bågar) är alltså frågan hur man ska lägga spänning på ingångarna så att utgången får spänning, och om det alls är möjligt. Formulera problemet som ett matematiskt optimeringsproblem. Tillhör problemet NP ? Tillhör det P ? Är det NP -fullständigt? Föreslå en lösningsmetod.

UPPGIFT 15.3

Betrakta följande två optimeringsproblem för riktade grafer.

Svar och lösningar

SVAR 15.1

- a) Hinkpackningsproblemet tillhör *NP* och är *NP*-fullständigt.
- b) LP-relaxationen av problemet ovan är lätt och tillhör *NP* och *P*.
- c) Kappsäcksproblemet tillhör *NP* och är *NP*-fullständigt.
- d) Hamiltonvägproblemet tillhör *NP* och är *NP*-fullständigt.
- e) Eulercykelproblemet tillhör *NP* och *P*. Kolla bara nodernas valens.
- f) Billigaste väg-problem: Sätt alla bågkostnader lika med 1. Finns en väg med målfunktionsvärde mindre eller lika med q ? Problemet tillhör *NP* och *P*.
- g) Problemet tillhör *NP* och är *NP*-fullständigt. Lägg till en nod med bara en anslutande bäge och sätt $q = n - 1$, så kan vi lösa Hamiltoncykelproblemet.
- h) Minkostnadsflödesproblemet tillhör *NP* och *P*.
- i) Valensbegränsade billigaste uppspännande träd-problemet (VMST) tillhör *NP* och är *NP*-fullständigt (med eller utan målfunktion).
- j) Nodpackningsproblemet (oberoende nodmängd) tillhör *NP* och är *NP*-fullständigt.
- k) Nodövertäckningsproblemet tillhör *NP* och är *NP*-fullständigt.
- l) Problemet tillhör *NP* och är *NP*-fullständigt. Man kan kolla vilka noder som är uppnåeliga från en eller flera noder polynomiskt, men sedan återstår ett mängd-övertäckningsproblem.

SVAR 15.2

Detta är *satisfieringsproblemet*, svagt förklätt, känt *NP*-fullständigt. Det kan lösas via omformulering till allmänt 0/1-problem (med t.ex. trädsökning).

SVAR 15.3

Både *P1* och *P2* tillhör *NP*. Ofta tillhör optimeringsversioner inte *NP*, men här vet vi att optimala målfunktionsvärdet ligger mellan noll och det totala antalet bågar i grafen. Vi kan därför kontrollera ja-svar för igenkänningsversionen för varje möjligt målfunktionsvärde och ändå inte använda mer än polynomisk tid. *P1* är polynomiskt lösbar (billigaste väg med bågkostnad ett). *P2* är *NP*-fullständigt. (Jämför med billigaste enkla väg när det finns negativa cykler.)