

## Laborationsinformation

# 1 NILEOPT: Nonlinear Optimization

## 1.1 Introduktion

NILEOPT är ett program för visuell optimering av olinjära optimeringsproblem (NLP). Huvudsyftet är att ge insikt i egenheter hos olinjära problem och metoder. NILEOPT innehåller två delar, en för problem av valfri dimension, med bivillkor, utan grafik, och en grafisk del för funktioner av två variabler, utan bivillkor. Programmet möjliggör inmatning av problemdata och förändring av data, samt lösning av problemet med koden Cobyla. Den grafiska delen av NILEOPT möjliggör visning av två-dimensionella funktioner (som 3D-yta och nivåkurvor) samt interaktiv applikation av ett flertal metoder för olinjär optimering utan bivillkor (se avsnitt 1.4). Programmet är skrivet i Tcl/Tk av Kaj Holmberg, och körs som telkit.

Programmet startas genom att öppna ett terminalfönster och sätta sökvägar genom att skriva `module add courses/TAOP88` och sedan skriva `junglebox-dine 3`.

NILEOPT minimerar en målfunktion, ev. under bivillkor. Alla bivillkor måste skrivas ut explicit. Man kan ha bivillkor med  $=$ ,  $\leq$  eller  $\geq$ . (NILEOPT vänder efter inmatningen  $\geq$ -bivillkor till  $\leq$ -bivillkor genom att multiplicera med  $-1$ .) Inmatning i editeringsfönstret är möjlig endast för polynom (utan parenteser), såsom  $f(x) = 3x_1x_2^2x_3^2 + 7x_1^4x_3 - 13x_1x_2 + 3x_3$ , men inte  $(x_1 - x_2)^2$ ,  $\sin(x)$  eller  $\exp(x)$ .

## 1.2 Meny

När NILEOPT startas, öppnas grundfönstret innehållande problemdel, grafisk del samt nedanstående menyer. Den första statusraden visar ett antal knappar samt problemets namn (indatafilens namn) och problemets storlek. Den andra raden anger funktionens namn. Tredje statusraden anger iterationspunkt, målfunktionsvärde samt aktuell metod.

**File** (läsa data/spara data/avsluta) "Alt"-f

Här kan man läsa in indata för ett problem samt spara data under aktuellt namn eller nytt (i internt format). Man kan även läsa in en modellfil, samt spara en sådan. Man kan skriva ut problem och grafik. Det går även att lista och radera befintliga problemfiler.

**Optimization** (optimering) "Alt"-o

Här kan man ange startpunkt för optimeringen. Man kan editera sitt problem, samt lösa problemet med Cobyla. (Detta kan också göras med knapparna `Edit` och `Opt` i statusraden.) Vid editering öppnas ett nytt fönster, som bl.a. innehåller knappen `Check`, som innebär att man kontrollerar hur resultatet blir om man skulle spara det

som just nu finns i editeringsfönstret. Gör gärna detta innan du sparar, eftersom de delar programmet inte förstår försvinner då man sparar.

#### **Visualization** (visning av problem) “Alt”-v

Här kan man välja att se problemet, samt gradient och Hessian till målfunktion och bivillkor (explicit och i given startpunkt) i problemfönstret. För grafiken kan man välja upplösning, hur och vad man ska se av funktionen, samt välja huruvida eventuella bivillkor ska ritas upp. Man kan även markera otillåtet område (lite oprecist).

#### **Graphic optimization** (grafisk optimering) “Alt”-g

Här kan man välja funktion och metod, samt stega fram metoden. (Det mesta av detta kan även göras med knapparna i statusraden.) Man kan även ändra maxgränsen för antal iterationer (som normalt är 1000).

#### **Help** (hjälp) “Alt”-h

Här kan man få upp kortare hjälptext i separat fönster, på svenska eller engelska. Man kan även lista existerande kortkommandon. Det finns möjlighet att ändra layout så att fönstret med nivåkonturerna blir större och 3D-grafen mindre.

På första statusraden finns knappar för ofta använda kommandon.

- EDIT: Editera aktuellt problem.
- STEP: Stega fram metoden ett steg. Förflyttning och ny iterationspunkt ritas upp i nivåkurvegrafen.
- RUN/STOP: Kör metoden utan att stanna varje iteration. (Alternativt: Hejda denna körning.)
- START POINT: Inmatning av startpunkt (kan vara nuvarande punkt, om en sådan finns).
- CLEAR PLOT: Rensa bort alla markerade punkter i nivåkurvegrafen.
- FUNCTION: Välj funktion ur listan.
- METHOD: Välj metod, dvs. hur sökriktningen beräknas. (Se metodlista nedan.)
- COBYLA: Lös aktuellt problem (med Cobylya).

Obsevera att man måste välja metod innan man trycker på START POINT. Om man kör COBYLA efterfrågas startpunkt, om en sådan ej finns. Annars används befintlig startpunkt. De andra metoderna startas genom att trycka på START POINT. Man kan även ange startpunkt genom att klicka i konturgrafen.

Till höger om problemfönstret finns knappar för att flytta och zooma grafikfönstren.

### **1.3 Dataformat**

NILEOPT erbjuder möjlighet att ändra data interaktivt, samt även att skapa ett nytt problem. NILEOPT sparar och läser problem på ett internt format som inte passar för manuell editering. Dessa filers namn slutar med “.nlp”.

NILEOPT kan dock läsa modeller i ett trevligare format. Detta format används för att skapa nya problem, och editera dem. Formatet är en förenklad version av vad GM-PL/AMPL använder. Förenklingen består av att enbart polynom kan läsas in, men inte trigonometriska funktioner, logaritmer och exponentialfunktioner. Dessutom accepteras inte parenteser, indexmängder, summor eller vektorer.

Formatet på datafilen är som följer. Varje rad avslutas med semikolon (;). Först deklaras varje variabel med

```
var x1;
```

Därefter ges målfunktionen med namn:

```
minimize Obj: 2*x1^2 - 2*x1*x2 + 3*x2;
```

Först `minimize` (eller `min`), sedan målfunktionens namn (här `Obj`), sedan ett kolon (:). Därefter följer själva funktionsuttrycket. Det består av ett valfritt antal termer med plus (+) eller minus (-) mellan. Varje term består av en koefficient samt variabler med multiplikationstecken (\*) mellan. Varje variabel kan ha en heltalig positiv exponent, som anges efter tecknet ^. (Ibland kan man behöva skriva mellanslag efter detta tecken för att få fram det.) Exponent lika med ett behöver ej skrivas in.

Sedan följer bivillkoren. Varje bivillkor inleds med texten `subject to` (eller `s.t.` eller `st`). Sedan följer bivillkorets namn (t.ex. `con1`), följt av ett kolon (:). Därefter följer ett funktionsuttryck, enligt samma regler som målfunktionen, därefter en likhet (=) eller olikhet (<= eller >=). Sista står högerledet (alltid till höger).

```
s.t. con1: x1^2 + x2^2 <= 1;
s.t. con2: x1 + x1*x2 <= 1;
st con3: x1+4*x2<=2;
```

NILEOPT kan också läsa och skriva modellfiler i detta format.

## 1.4 Grafisk optimering

Den grafiska optimeringsmiljön i NILEOPT kan hantera problem i två variabler. (Om problemet har fler än två variabler, ritas inget ut.) Den innehåller ett fönster med en 3D-bild av funktionsytan och ett fönster med nivåkurvor.

Huvudfunktionen är att göra optimering stegvis och visa varje iterationspunkt i nivåkurvegraf. Man kan därmed jämföra olika metoder, och även effekten av olika startpunkter.

Denna del av NILEOPT kan läsa in mer generella funktionsuttryck, men om de inte uppfyller kraven vid inmatningen som beskrivs i föregående stycke (polynom utan parenteser), kan inte editering göras. Det finns ett stort antal testfunktioner. De kan delas in i tre kategorier:

1. Polynom enligt reglerna. Dessa kan sparas i internt format, editeras, och gradient och Hessian kan beräknas symboliskt (exakt). De betecknas med \* i menyn.
2. Mer generella funktioner, där man har specificerat explicit gradient och Hessian i indatafilen. Dessa funktioner kan inte editeras eller sparas i internt format, men man kan använda metoder som kräver gradient och Hessian. De betecknas med + i menyn.
3. Mer komplicerade funktioner, för vilka gradient och Hessian inte kan beräknas eller har angetts. De betecknas med - i menyn. Dessa funktioner anges av en subrutin (i Tcl) som ger funktionsvärdet. Denna subrutin kan vara hur komplicerad som helst. (Det är inte tänkt att användaren ska skapa dylika subrutiner, så inga instruktioner ges för detta här.)

NILEOPT innehåller möjligheten att numeriskt beräkna en approximation av gradienten. Därför kan gradientbaserade metoder användas. Någon approximation av Hessianen finns dock inte tillgänglig.

Varje gång en funktion väljs, visas funktionsuttrycket (om det är tillgängligt) i problemfönstret. Om en funktion av typ 1 har valts, kan den bearbetas vidare i editeringsfönstret. När man väljer funktion, kan man välja funktion 0, vilket är målfunktionen i det problem som finns i problemfönstret, som man själv kan ha matat in.

## Metoder:

**Cobyla:** Koden Cobyla (Constrained Optimization by Linear Approximation).

**Steepest descent:** Brantaste lutningsmetoden.

**Newton's method:** Newtons metod, med explicit, exakt gradient och Hessian.

**Levenberg (convexified Newton):** Levenbergs konvexifiering av Newtons metod, med explicit, exakt gradient och Hessian.

**Levenberg-Marquardt (convexified Newton):** Levenberg-Marquardts konvexifiering av Newtons metod, med explicit, exakt gradient och Hessian.

**Nelder-Mead's metod:** Nelder-Meads metod (med en vippande simplex). Kräver ej gradient.

**Conjugate gradient (Fletcher-Reeves):** Modifierad brantaste lutningsmetod, enligt Fletcher-Reeves.

**Conjugate gradient (Polak-Ribiere):** Modifierad brantaste lutningsmetod, enligt Polak-Ribiere.

**Conjugate gradient (Hestenes-Stiefel):** Modifierad brantaste lutningsmetod, enligt Hestenes-Stiefel.

**Quasi Newton (DFP):** Newtons metod med Hessianen ersatt av en approximation, enligt Davidon-Fletcher-Powell. Kräver ej Hessian.

**Quasi Newton (Broyden):** Newtons metod med Hessianen ersatt av en approximation, enligt Broyden. Kräver ej Hessian.

**Quasi Newton (SR1):** Newtons metod med Hessianen ersatt av en approximation, enligt "Symmetric Rank 1". Kräver ej Hessian.

**Quasi Newton (BFGS):** Newtons metod med Hessianen ersatt av en approximation, enligt Broyden-Fletcher-Goldfarb-Shanno. Kräver ej Hessian.

NILEOPT stannar då gradientens norm är mycket liten, då förflyttningen blir väldigt liten eller då maxantal iterationer har gjorts. (Maxantal iterationer kan ändras.)

Vid linjesökningen beaktas inte bivillkoret  $t \geq 0$ , utan en negativ steglängd kan fås. (Detta bör dock vanligtvis inte spela någon roll.) Om flera lokala minima finns längs sökriktningen, är det inte säkert att det närmaste eller bästa fås.

Om man kör COBYLA efterfrågas startpunkt, om en sådan ej finns. Annars används befintlig startpunkt. De andra metoderna startas genom att trycka på START POINT eller klicka i konturgrafan, vilket betyder att man måste välja metod innan man väljer startpunkt.