Ottimizzazione dei Sistemi Complessi

A.A. 2015-16 – 14 Febbraio 2017

prova d'esame

1. (**8 punti**) Dato il problema multiobiettivo seguente

$$\begin{aligned} &\min\ (x_1-1)^2+(x_2-1)^2; (x_1+1)^2+x_2^2 \\ &s.t.\ x_1+x_2 \le \tfrac{1}{2} \\ &\quad\ x_1^2+x_2^2 \le \tfrac{1}{4} \end{aligned}$$

   - aiutandosi anche con una rappresentazione grafica del problema, determinare il vettore ideale degli obiettivi z_{id} e i valori delle variabili di decisione che lo determinano;
   - scrivere il problema che si ottiene applicando il metodo degli ϵ-vincoli e mettendo a vincolo la seconda funzione obiettivo;
   - determinare un ottimo di Pareto risolvendo il problema degli ϵ-vincoli con $\epsilon_2 = 2$.

2. (**8 punti**) Si consideri il seguente problema vincolato:

$$\begin{aligned} &\min\ (x_1-1)^2+x_2^2+2x_1x_2 \\ &s.t.\ x_1^2+x_2^2 \le \tfrac{1}{4} \\ &\quad\ x_1+x_2 \le \tfrac{1}{2} \end{aligned}$$

   Con riferimento al problema di sopra,

   - scrivere l'espressione della funzione Lagrangiana del problema;
   - scrivere l'espressione di una funzione di Penalità sequenziale esterna;
   - scrivere l'espressione di una funzione di Penalità interna (o di barriera);
   - scrivere l'espressione di una funzione Lagrangiana aumentata (sequenziale);

3. (**8 punti**) Dato il problema di controllo ottimo a tempo continuo con, con T fissato:

$$\begin{aligned} &\min\ \frac{1}{2}\int_0^T \left(x_1(t)^2+x_2(t)^2+u(t)^2\right) dt \\ &\dot{x}_1(t) = x_2(t) \\ &\dot{x}_2(t) = x_3(t) \\ &\dot{x}_3(t) = -x_1(t) - x_2(t) + cos(u(t)) \\ &x_1(0) = x_2(0) = x_3(0) \\ &x_1(T) = x_2(T) = 1 \\ &-\tfrac{\pi}{2} \le u(t) \le \tfrac{\pi}{2} \end{aligned}$$

   - (**3 punti**) scrivere le condizioni necessarie di ottimalità;
   - (**2 punt1**) dire in quale caso dovrebbe valere anche la condizione

$$-\left(1+\frac{u(T)^2}{2}\right) + \lambda_1(T) + \lambda_2(T)x_3(T) = 0$$

     motivando la risposta;
   - (**3 punti**) scrivere il problema di ottimizzazione nonlineare che si ottiene con una discretizzazione rispetto al tempo con $\Delta t = \frac{T}{1000}$, determinandone le dimensioni (numero di variabili e di vincoli).

segue dietro ...

4. (**8 punti**) Riordinare il seguente script `Julia`

`NMx = cell(0)` `NMf = Array(Float64,0)`	**1**
`problems = cell(0)` `push!(problems,(McKinnon,2))`	**2**
`push!(NMx,res.minimum)` `push!(NMf,res.f_minimum)`	**3**

`for i in 1:50`
`start = -10+20*rand(n)`
`res = optimize(objfun,start,NelderMead(iterations=10000));`
	4

`else`
`return 15*x[1]^2 + x[2] + x[2]^2`
`end`
`end`
	5

`for opt_case in problems`
`objfun = opt_case[1]`
`n = opt_case[2]`
	6

`function McKinnon(x::Array)`
`if x[1] <= 0.0`
`return 150*x[1]^2 + x[2] + x[2]^2`
	7

`using Optim` `srand(1)`	**8**

`println(i," ", NMf[end])`
`end`
`end`
	9