|  |  |  |
| --- | --- | --- |
| Veränderung | Ergebnis | Parameter |
| Nu = 0 immer so gewählt, G aus nu und d berechnet  Mit fminimax | Ohne variables J (klar): 64 Iterationen; ca 4 Sekunden; f:hat 0.0095, E > 0; genau jeder zweite Schritt null-step; gegen Ende sehr geringe Änderung des Funktionswertes; kein Springen von x um Minimum, x wird nicht wieder größer  X0 = -5: alles genau wie für +5 | m = 0.5; gamma = 0.01, tol = 10^-6 kmax = 100, x0 = 5/-5  t = 1, null-step: 0.7\*t |
| Nu = 0 immer so gewählt, G aus nu und d berechnet  Mit IP / Active Set | Ohne varibles J:  AS: 4 Iterationen; ca 0.1 Sekunden; f\_hat = 0.00000025; E < 0 in serous step IP: 4 Iterationen; ca 0.2 Sekunden; f\_hat = 0.00000025; E < 0 in serous step  X0 = 10, 20: Ergebnisse sehr ähnlich; AS: x0 = -5: 93 Iterationen; ca. 3 Sekunden; f\_hat = 1.11; E < 0; anfangs mehr null-steps, Ende jeder zweite Schritt null-step, kein Springen von x aber sehr wenig Änderung IP/AS haben ein Problem mit negativem Starwert!!! Aber x-Werte in Algo bei positivem Startwert zwischendrin negativ…  Für „kleinere“ Startwerte (-1 bis -4) nur ca. eine Iteration, sinnlose Ergebnisse  Mit variablem J: tritt das Springen von x auf?! Werden falsche Indizes entfernt? | Wie oben  Außer Startwert |
| G aus Lagrange-Multiplikatoren  Mit IP / Active Set | Ohne Anpassung von J: AS: kann nicht beenden; ca. 3 Sek; E1 nie = E2; nur Null-steps, x geht passend runter, aber nie serious step; E1, E2 immer > 0; c immer > 0; Summe alpha geht immer weiter von 1 weg, nach ca. 65 Schritten einfach 0 IP: kann nicht beenden; ca. 9 Sek; ; E1 nie = E2; nur Null-steps; , x geht passend runter, aber nie serious step; alles immer passend > 0; Summe alpha geht immer weiter von 1 weg, nach ca. 55 Schritten einfach 0  X0 = -3: AS: alpha-Werte passen, sehr wenig Iterationen aber völlig falsches Ergebnis IP: wie bei AS; auch x erst runter dann hoch x0 = -5: AS: wie in Nu = 0; alphas sind länger ungefähr 1  IP: E1 kleiner 0; schafft nicht; 8 Sekunden; alpha am Ende = 0; E‘s nie gleich; Ende ca jeder 2. Schritt null-step; Ergebnis trotzdem schlecht; x erst runter, dann hoch und bleibt da stecken  Mit E2 für delta: AS: exakt identisches Ergebnis  IP:ebenso Mit Anpassung von J: AS: wie in IP, nur alpha irgendwann = 0 und E1/2 immer > 0 IP: mit E1: schafft nicht; 8 Sekunden; E1 immer ungl. E2; nur null-steps; f geht sinnvoll runter, hängt dann bei 2,5\*10^-7 (wie schon bei anderen Konfigurationen?); alpha bis 55 Schritte etwa 1, wird immer schlechter E1 unur im ersten Schritt kleiner 0  X0 = -5: AS: 0,2 Sek; 6 Iters; 2 null-steps; x springt erst von gutem Wert weg, dann ganeu da wieder hin; E’s nie gleich; alphas ung. 1; E1 zweimal kleiner 0, korreliert nicht mit null-steps (?) IP: wie AS, aberE’s immer > 0;  X0 = -3: IP: unsinniges Ergebnis; 4 Iters; 0,2 Sek; x springt von gutem Wert weg; AS: mehr oder weniger genauso | Wie oben |
| E aus Lagrange-Multiplikatoren vs. E in „langer Form“ | Exakt gleiche Ergebnisse wenn nur in delta eingesetzt; Berechnungszeit? Wahrscheinlich egal |  |
| Unterschiedliche Startwerte | Siehe oben |  |
| t ändern |  |  |
| J dynamisch vs. Alles hinzufügen (IP/AS) | Springen von x tritt auf? Siehe oben |  |
| Suchrichtung d nach oben und unten begrenzt durch D | X nimmt immer größtmöglichen Suchrichtungswert an??? |  |
|  | prox\_bundle\_nonconv(5, par, abl\_par, 1,1,1)  mit D beschränkt durch +-10 und Abbruchbedingung auch wirklich mit f\_(k+1) und nicht was anderes wie vorher 🡪 c kleiner 0 aufgetreten |  |
| Mit quadprog | Alpha immer genau 1 aber noch kein sinnvolles Ergebnis (kommt bei Startwert raus), macht nur Null-steps, ändert x nur im ersten Schritt (in die falsche Richtung), dann nicht mehr Version vom 11.07. 🡪 f\_k in f\_k+1 bei Abbruchbedingung geändert |  |
| D nicht mehr | IP: Alpha nicht mehr ganz genau 1 aber sehr gut  AS: Alpha etwas schlechter, aber immer noch sehr gut  Erste beide mit nicht variablem J Mit variablem J: AS: alphas deutlich schlechter, aber noch bei 1 (nicht wie bei fmincon, wo völlig falsch) IP: sehr sehr genau 1 |  |