

Komputeralgebra rendszerek

XI. A megoldók

Czirbusz Sándor
czirbusz@gmail.com

Komputeralgebra Tanszék
ELTE Informatika Kar

2010-2011 őszi

Index I

1 „Egyszerű” egyenletmegoldó

- Maple
 - A solve()
 - Használata
- Sage
 - Ez is solve()

2 Numerikus megoldások

- Maple
 - Az fsolve()
 - Szerszámok
- Sage
 - A find_root()

3 Egyéb megoldók a Maple-ban

- Diofantikus problémák
- Kongruenciák

Index II

- Rekurziók

- 4 • Egyéb megoldók a Sage–ban

- Kongruenciák
- Egyenlőtlenségek
- Kínai maradékolás

Egzakt megoldások

solve

- Szintaxis : **solve(egyenletek, változók)**
- Az első paraméter lehet egyetlen egyenlet, egyenlőtlenség, vagy ezek listája|halmaza.
- A második paraméter opcionális; lehet egyetlen változó, vagy változók listája|halmaza
- Az output struktúrája a megadás struktúrájától függ: ha valamelyik összetett, a megoldáselemek halmazok
- Az output szekvencia
- A megoldás felhasználása máshol : az **assign()** segítségével

Tippek és trükkök I

- Ha egyértelmű, nem kellene a változók
- Nem kötelező egyenlet jobboldalát megadni, ha az zérus
- Ismeretlenek lekérdezése : **indets**
- Magasabbfokú polinomok megoldása *RootOf*–alakú, néha használható a **convert(radical)**
- Páratlan gyökök : **surd()** !
- Valós gyökök kiválasztása vagy **select()** segítségével, vagy valósban megoldani a
use RealDomain in ...end use struktúrával

Tippek és trükkök II

- Van amikor elegendő az **eliminate()** átrendező
- **_EnvAllSolutions := true** pl. trigonometrikus egyenletek
- **_SolutionsMayBeLost** állapot : valami miatt nem oldható meg (jobb, mintha nem szól ☺)
- Ellenőrzés : legegyszerűbb a **subs**–segítségével.
- Lásd help

Egy „kisebb” megoldó

solve

- Szintaxis : **solve(egylist, varlist[, solution_dict])** (vagyis a paraméterek listák)
- Az opcionális paraméter default értéke false. Ha true, szótárak (dictionary) listáját kapjuk
- Ez utóbbi esetben az eredmény „továbbhasznosítható”
- A megoldás szimbolikus, ha lehetséges, de egyenletrendszereknél lehet approximált
- Egyszerűsíteni itt lehet, ha nulla a jobboldal, illetve ha egyértelműek a változók.
- Csak egyenletre jó !

Csak közelítőleg 😊

fsolve

- Szintaxis : **fsolve(equations [,var][,opciók])**
- Az egyenletekre és a változóra a **solve** szabályai vonatkoznak.
- Az opciók:
 - *complex* – komplex megoldások is
 - *avoid=sols* – már megtaláltakat ne keressen (halmaz)
 - $x_0 \dots x_1$ – hol keresse

Hol keressük?

Tools

- **realroot(p[,hossz])** : hova esnek „p” valós gyökei. Az eredmény listaként megadott intervallumok listája.
- **sturm(p,var, l)** : hány valós gyöke van egy intervallumban. Az „l” lehet **realroot** kimenete
- **starting_values** : az iteráció kezdőértéke
- **fulldigits** : a „Digits” rendszerérték szerinti pontossággal

Gyökkereső

`find_root`

- Szintaxis : **`find_root(f, a, b, xtol, rtol, maxiter, full_output)`**
f, a, b : az egyenlet és az intervallumok
xtol, rtol :
maxiter : default 100
full_output : info a konvergenciáról is
- A **scipy** csomag segítségével többféle optimalizációra képes. Betöltése :
`from scipy import optimize`

isolve

isolve

- Szintaxis: **isolve(eqns[, vars])**
- A megoldásokat a **_Zn** változókkal generálja, kivéve, ha van második paraméter

msolve

- Szintaxis :**msolve**(egyenletek [, vars], m)
- Az „m” a modulus
- A „vars” szerepe ugyanaz, mint az **isolve** esetén
- A kínai maradékolója a **chrem**(ulist, mlist) függvény
- Az „ulist” nemcsak egészeket tartalmazhat, hanem polinomokat is
- A **numtheory** csomag kínai maradékolója a **mcombine**

rsolve

- Szintaxis : **rsolve(egys, fns[, mode])**
- Az „egys” a rekurzió(ka)t és a kezdetiértékeket jelenti
- A „fns” amire meg kell oldani
- A „mode” lehet
 - **genfunc** – genrátorfüggvényhez
 - **makeproc** – egy Maple procedúrát kapunk
 - **series** – formális hatványsor

solve_mod

- Szintaxis : **solve_mod(eqns, modulus, solution_dict=False)**
- Az egyenletek polinomok, az adott maradékosztályban oldja meg
- „solution_dict” – szótár a megoldás
- Összetett modulust faktorizálja, az eredményt kínai maradékkal számolja
- Nagy modulusra nem hatékony

solve_ineq

- A Maximán keresztül oldja meg az egyenlőtlenségeket

crt

Több függvény a kínai maradé problémára

- **crt(ulist, mlist)** – lényegében azonos a Maple-függvénnyel, működik polinomokra is
- **CRT_list ((ulist, mlist))** – ugyanaz, csak egészekre
- **CRT_basis(mlist)** – a szorzat-gyűrű egy bázisa
- **CRT_vectors(X, mlist)** – több kínai maradékproblémára ugyanazon modulusokkal; az „X” vektor. (Hatékonyabb, mint többször lefuttatni)

Példák

Lásd a munkalapokon