ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ v PRAZE

Fakulta elektrotechnická

Katedra řídící techniky

Přehled dostupného software pro řešení soustav polynomiálních rovnic.

Bakalářská práce

Vedoucí bakalářské práce: Ing. Petr KUJAN
Oponent bakalářské práce: Ing. Jan KELBEL
Studijní program: Elektrotechnika a informatika
Obor: Kybernetika a měření

2007 Petr Štefan

ČVUT Praha

České vysoké učení technické v Praze, fakulta elektrotechnická

Katedra řídicí techniky | Školní rok: 2006/2007

Zadání bakalářské práce

Student: Petr Štefan

Obor: Kybernetika a měření

Název tématu: Přehled dostupného software pro řešení soustav polynomiálních rovnic

Zásady pro vypracování:

Cílem práce je vytvořit přehled volně dostupného software na řešení soustav polynomiálních rovnic. Tato úloha je obecně velmi komplikovaná a je známa celá řada různých metod a přístupů k řešení (eliminační metody, Groebnerovy báze, numerické iterativní metody, optimalizace a další), které se liší různou efektivitou v závislosti na řešené úloze.

Cílem je naučit se ovládat nalezený software, stručně popsat (zařadit) metodu výpočtu a otestovat jej na konkrétní úloze hledání spínací sekvence pro vícehladinový konvertor stejnosměrného napětí na střídavé, které vede právě na řešení soustavy polynomiálních rovnic. Dosažené výsledky (rychlost a přesnost výpočtu) by měli být navzájem porovnány a vyhodnoceny.

Seznam odborné literatury:

FRISCO - A Framework for Integrated Symbolic/Numeric Computation, on-line: ww.nag.co.uk/projects/FRISCO.html

Vedoucí bakalářské práce: Ing. Petr Kujan

Datum zadání bakalářské práce: zimní semestr 2006/07

Termín odevzdání bakalářské práce: 15. 8. 2007

Prof. Ing. Michael Šebek, DrSc.
vedoucí katedry

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE Fakulta elektrotechnická L.S. -4-

Prof. Ing. Zbyněk Škvor, CSc.
děkan

V Praze, dne 6. 3. 2007

1. Prohlášení

Prohlašuji, že jsem svoji bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literatura, projekty, SW, …) uvedené v přiloženém seznamu.

V Liberci dne 14. 8. 2007 ……Petr Šlégr……

podpis

2. Poděkování

Rád bych poděkoval vedoucímu mé práce, Ing. Petru Kujanovi, za odborné připomínky, trpělivost a pomoc při zpracování této práce, svým přátelům a rodinným příslušníkům za různé připomínky, zejména v oblasti slovosledu, možnostech ve vyjadřování a celkové grafické úpravě tohoto textu.

3. Anotace

3.1. Anotace

Název práce:	**Přehled dostupného software pro řešení soustav polynomiálních rovnic.**
Autor:	Štefan Petr
Katedra:	Katedra řídící techniky
E-mail:	stafap1@fel.cvut.cz
Vedoucí bakalářské práce:	Ing. Kujan Petr

Abstrakt: Cílem práce je vytvořit přehled volně dostupného softwaru na řešení soustav polynomiálních rovnic, naučit se ovládat nalezený software, stručně popsat (zařadit) metodu výpočtu a otestovat jej na konkrétní úloze hledání spínací sekvence pro vícehladinový konvertor stejnosměrného napětí na střídavé. Dosažené výsledky by měli být navzájem porovnány a vyhodnoceny.

Klíčová slova: soustava polynomiálních rovnic, software, matematické metody

3.2. Annotation

Title:	**Survey available software tool for solving the set of polynomial equations.**
Author:	Štefan Petr
E-mail:	stefap1@fel.cvut.cz
Department:	Department of Control Engineering
Supervisor:	Ing. Kujan Petr

Abstract: The aim of my work is to make survey in freely available software for solving systems of polynomial equations, to teach how to use them, to make a brief description of them, to test them on a concrete polynomial system and finally co compare the results.

Keywords: system of polynomial equations, software, numerical methods

4. Obsah

1. Prohlášení ... 2
2. Poděkování ... 3
3. Anotace ... 4
3.1. Anotace ... 4
3.2. Annotation ... 4
4. Obsah ... 5
5. Úvod ... 7
6. Vícehladinový konvertor ... 8
7. Přehled softwaru a jeho popis ... 15
7.1. Maple 11 ... 15
7.1.1. Popis programu ... 15
7.1.2. Instalace ... 17
7.1.3. Syntaxe ... 17
7.2. Mathematica 5.2 ... 18
7.2.1. Popis programu ... 18
7.2.2. Instalace ... 19
7.2.3. Syntaxe ... 19
7.3. MATLAB ... 20
7.3.1. Popis ... 20
7.3.2. Historie ... 22
7.3.3. Instalace ... 22
7.3.4. Syntaxe ... 22
7.4. MuPAD ... 23
7.4.1. Popis programu ... 23
7.4.2. Instalace ... 24
7.4.3. Syntaxe ... 24
7.5. Gloptipoly 2 ... 25
7.5.1. Popis programu ... 25
7.5.2. Instalace ... 26
7.5.3. Syntaxe ... 26
7.6. Maxima ... 27
7.6.1. Popis programu ... 27
7.6.2. Historie ... 28
7.6.3. Instalace ... 29
7.6.4. Syntaxe ... 29
7.7. Yacas ... 30
7.7.1. Popis programu ... 30
7.7.2. Instalace ... 31
7.7.3. Syntaxe ... 31

7.8. PHCpack ... 32
7.8.1. Popis ... 32
7.8.2. Instalace ... 33
7.8.3. Syntaxe ... 34
7.9. SINGULAR ... 34
7.9.1. Popis programu ... 34
7.9.2. Historie ... 35
7.9.3. Instalace ... 36
7.9.4. Syntaxe ... 36
7.10. OCTAVE ... 37
7.10.1. Popis programu ... 37
7.10.2. Historie ... 38
7.10.3. Instalace ... 39
7.10.4. Syntaxe ... 39
8. Další nalezený software ... 40
9. Testování na konkrétní úloze ... 41
9.1. Konfigurace použitého počítače ... 41
9.2. Výsledky jednotlivých testů ... 42
9.2.1. Mathematica ... 42
9.2.2. Matlab ... 42
9.2.3. Gloptipoly ... 42
9.2.4. Octave ... 42
9.2.5. PHCpack ... 43
9.2.6. Singular ... 43
9.2.7. MuPAD ... 44
9.2.8. Maple ... 44
9.2.9. Maxima ... 44
9.2.10. Yacas ... 44
9.3. Porovnání výsledků ... 45
10. Závěr ... 47
11. Použité zdroje ... 48
11.1. Literatura ... 48
11.2. Elektronické zdroje ... 48
12. Příloha A – seznam obrázků ... 50
13. Příloha B – seznam tabulek ... 51
14. Příloha C – obsah přiloženého CD ... 52
15. Příloha D – výstup programu Singular ... 53
16. Příloha E – výstup programu Mathematica ... 60

5. Úvod

Cílem této práce je vytvořit alespoň stručný přehled volně dostupného softwaru pro řešení soustav polynomiálních rovnic. S nalezenými programy se naučit pracovat, otestovat je a alespoň stručně popsat.

V počátcích práce na tomto textu jsem se potýkal s jistými problémy. Zejména se jednalo o to, že žádný z nalezených programů není český, tudíž jsem byl nucen vyhledávat v anglickém jazyce. Po prvních neúspěšných pokusech vyhledávat klíčová slova „set of polynomial equations“, kde jsem použil pro „soustavu“ překlad „set“, jsem zjistil, že mnohem více se toho skrývá pod překladem „system“. Nejlepších výsledků ve vyhledávání jsem dosáhl pod klíčovými hesly „polynomial system solver“.

Na internetu existuje spousta projektů a knihoven pro práci s polynomiálními soustavami, avšak ne všechny jsou dostatečně zdokumentované, tudíž některé nebylo možné otestovat. Některé z těchto aplikací se zdály být efektivní a použitelné, když jsem je testoval na jedné z jednodušších soustav, avšak při testování na složitější soustavě vykazovaly značné nedostatky.

V začátku celé své práce bych chtěl nastínit problém hledání spínací frekvence pro vícehladinový konvertor stejnosměrného napětí na střídavé, alespoň stručně popsat metodu návrhu, která vede právě na řešení pomocí soustav polynomiálních rovnic. Tyto rovnice jsou velice komplikované a triviálními metodami takřka nespočítatelné. Proto software, který budu testovat, využívá speciálních algoritmů k jejich výpočtu.

V další části práce budu popisovat jednotlivý nalezený software. Stručně nastíním algoritmus, pomocí kterého dochází k výpočtu, otestuji daný software na konkrétní úloze a to nejdříve na jednodušší soustavě a následně na složitějších. Nalezené programy se také budu snažit nějak klasifikovat a rozdělit podle typu licence, podle toho, zda pracují samy nebo zda k jejich běhu je potřeba další software. V začátku této části bych chtěl také zahrnout některé aplikace, které nespadají do kategorie freeware, nicméně pro tyto aplikace disponuje ČVUT multilicencí, tudíž je lze používat na některých počítačích ve škole, některé dokonce i doma.

V závěru práce se pokusím všechny nalezené programy porovnat, a to z hlediska rychlosti a přesnosti výpočtu, složitosti instalace a následné manipulace.

6. Vícehladinový konvertor

Konvertor je elektrické zařízení, které je schopno přeměnit napájecí napětí některých stejnosměrných zdrojů (baterie, solární články, atd.) na střídavé napětí.

V současné době má velice široké využití. Například výstup solárních článků je stejnosměrné napětí. Pokud chceme takovéto zařízení zapojit do stejnosměrné sítě, potřebujeme pro to nějaké další rozhraní. Vícehladinový konvertor je pro to ideálním řešením. Pro tyto situace se výborně hodí beztransformátorový vícehladinový invertor kvůli svým vysokým jmenovitým hodnotám výkonu. Vícehladinové invertory pracují s vyšší účinností, protože se dokáží přepínat s mnohem menší frekvencí než u invertorů založených na pulsně - šířkové modulaci (PWM). Dalším důležitým příkladem jsou hybridní vysoko výkonová elektrická vozidla (hybrid electric vehicles – HEVs), které mají obrovský elektrický pohon, jako jsou přívěsy traktorů, překládací zařízení nebo vojenská vozidla. Tyto přístroje potřebují vyspělé elektrické invertory, aby splnily své vysoké nároky na napájení (vice než 100kW). Vynalezení takovýchto hnacích jednotek bude mít za následek vysoké úspory paliva, snížení emisí a také lepší vlastnosti vozidla (zrychlení a brzdění) .

Elektrické schéma takovéhoto zařízení je uvedeno na obrázku 1, schodovité výstupní napětí je na obrázku 2.

Fourierova řada pro stupňovitý výstupní signál (viz obrázek 2) vypadá následovně:

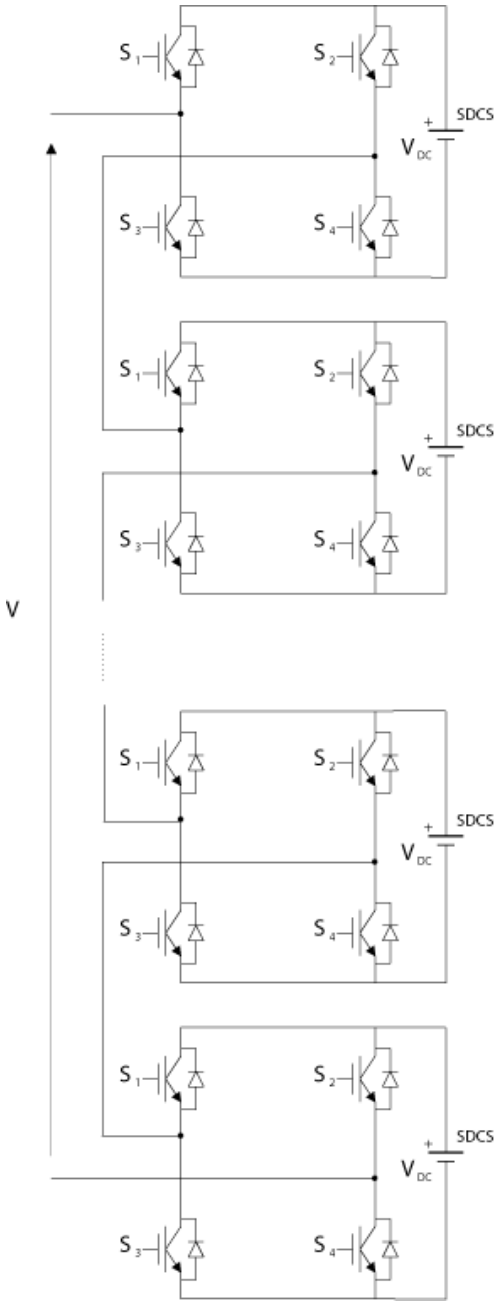
$$V(\omega t) = \sum_{k=1,3,5,\ldots}^{\infty} \frac{4V_{DC}}{n\pi} \cdot \left(\cos(n\omega\theta_1) + \cos(n\omega\theta_2) + \ldots + \cos(n\omega\theta_s)\right) \cdot \sin(n\omega t), \qquad (1)$$

kde VDC je napětí stejnosměrného zdroje (předpokládá se, že všechny zdroje mají stejné napětí), s je počet stejnosměrných zdrojů a ω je požadovaná výstupní frekvence. Pro neznámé spínací úhly (viz obrázek 2) musíme dodržet:

$$0 < \theta_1 < \theta_2 < \ldots < \theta_s < \frac{\pi}{2\omega}. \qquad (2)$$

Derivování vztahu (1) pro jeden zdroj je následující. Je to lichá periodická funkce (viz obrázek 3), proto pro jeden krok Fourierovy řady je:

$$V^{(1)}(\omega t) \sim \sum_{k=1}^{\infty} b_k \sin(k\omega t),$$



Obrázek 1: Jednofázová struktura vícehladinového kaskádního H-můstkového invertoru

kde

$$b_k = \frac{4}{T}\int_0^{\frac{T}{2}} V(t)\sin(k\omega t) = \frac{4}{T}\int_{0-\theta_1}^{\frac{T}{2}-\theta_1} V_{DC}\sin(k\omega t)$$
$$= \frac{4}{T}V_{DC}\frac{1}{k\omega}\left(-\cos\left(k\omega\left(T/2-\theta_1\right)\right)+\cos\left(k\omega\theta_1\right)\right). \quad (3)$$

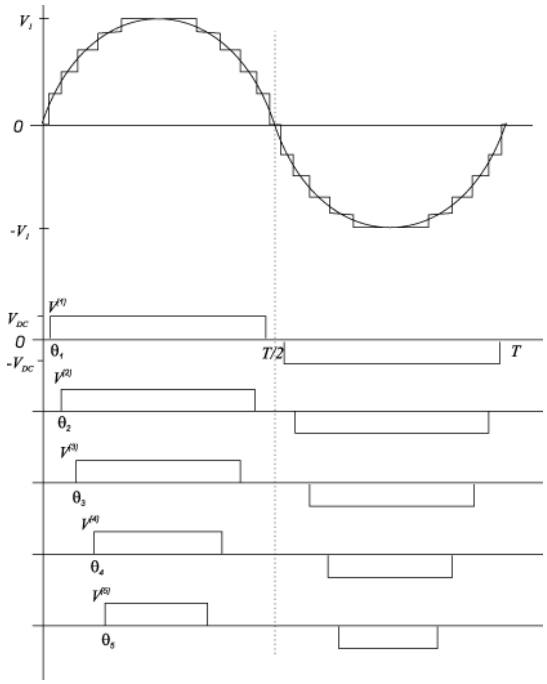
Výraz v závorce v rovnici (3) může být upraven

$$-\cos\left(k\omega\left(T/2-\theta_1\right)\right)+\cos\left(k\omega\theta_1\right)= \quad (4)$$

$$= -\cos(k\omega T/2)\cos(k\omega\theta_1) - \sin(k\omega T/2)\sin(k\omega\theta_1) + \cos(k\omega\theta_1) = \\ = \begin{cases} 0 & pro\ k = 0, 2, 4, \ldots \\ 2\cos(k\omega\theta_1) & pro\ k = 1, 3, 5, \ldots \end{cases} \tag{5}$$

V posledním výrazu byla použita rovnost $\omega = \frac{2\pi}{T}$. Vše dohromady dává

$$V^{(1)}(\omega t) \sim \sum_{k=1,3,5\ldots} \frac{4V_{DC}}{k\pi}\cos(k\omega\theta_1)\sin(k\omega t). \tag{6}$$



Obrázek 2: Výstupní průběh 11-hladinového kaskádního invertoru

Pro další zdroj se předchozí procedura opakuje, jen s tím rozdílem, že se použije úhel θ_2 ($\theta_2 > \theta_1$), tj. můžeme psát

$$V^{(2)}(\omega t) \sim \sum_{k=1,3,5\ldots} \frac{4V_{DC}}{k\pi}\cos(k\omega\theta_2)\sin(k\omega t). \tag{7}$$

Pro všechny další zdroje je postup stejný, Respektive výsledky $V^{(i)}(\omega t)$ pro různé úhly se sčítají

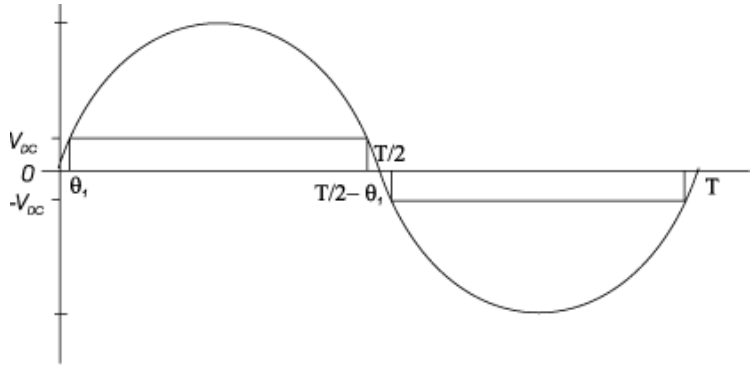
$$V(\omega t) = \sum_{i=1}^{s} V^{(i)}(\omega t)$$

a dostáváme přesně (1).

Výpis konkrétních lichých harmonických je následující

$$V(\omega t)=\sum_{i=1}^{s}V^{(i)}(\omega t)=$$
$$=\sum_{k=1,3,5,,,}\frac{4V_{DC}}{k\pi}\cdot\left(\cos(k\omega\theta_1)+\cos(k\omega\theta_2)+\ldots+\cos(k\omega\theta_s)\right)\cdot\sin(k\omega t)=\sum_{k=1,3,5,,,}V_k\sin(k\omega t)=$$
$$=\frac{4V_{DC}}{1\pi}\cdot\left(\cos(\omega\theta_1)+\cos(\omega\theta_2)+\ldots+\cos(\omega\theta_s)\right)\cdot\sin(\omega t)+$$
$$+\frac{4V_{DC}}{3\pi}\cdot\left(\cos(3\omega\theta_1)+\cos(3\omega\theta_2)+\ldots+\cos(3\omega\theta_s)\right)\cdot\sin(3\omega t)+$$
$$+\frac{4V_{DC}}{5\pi}\cdot\left(\cos(5\omega\theta_1)+\cos(5\omega\theta_2)+\ldots+\cos(5\omega\theta_s)\right)\cdot\sin(5\omega t)+$$
$$+\ldots$$

V předchozích rovnicích $V^{(i)}(\omega t)$ je Fourierova řada pro jeden zdroj (viz rovnice (6) a (7)) sepnutý po úhlem θ_i a V_k je amplituda k–té harmonické.



Obrázek 3: Výstupní průběh jednoho stejnosměrného zdroje

Stupňovitý výstupní signál je ještě dále filtrován. Proto jsou spínací úhly nejlépe nastaveny tak, že první harmonická výstupního schodovitého signálu bude rovna požadovanému výstupnímu signálu $V_1 \sin(\omega t)$ a několik následujících bude nulových. Zbývající vyšší harmonické budou odstraněny filtrem. Podle aspektu aplikace předpokládáme třífázové zapojení. (Jednofázové zapojení může být řešeno analyticky a také jednodušeji pomocí Newtonovy identity (Chudnovsky and Chudnovsky, 1999; Czarkovski et al., 2002).) Následně jsou harmonické, které jsou násobkem tří, automaticky vynulovány systémem a proto není zapotřebí, aby byly rovné nule. Pokud vynecháme tyto harmonické, pak jsme schopni vynulovat větší počet harmonických, tudíž výstupní signál méně osciluje. Z těchto požadavků a rovnice (1) vyplývá následující transcendentní soustava rovnic

$$\begin{aligned}
&\frac{4V_{DC}}{1\pi}\cdot\left(\cos(\omega\theta_1)+\cos(\omega\theta_2)+\ldots+\cos(\omega\theta_s)\right)=V_1\\
&\left(\cos(5\omega\theta_1)+\cos(5\omega\theta_2)+\ldots+\cos(5\omega\theta_s)\right)=0\\
&\left(\cos(7\omega\theta_1)+\cos(7\omega\theta_2)+\ldots+\cos(7\omega\theta_s)\right)=0\\
&\qquad\qquad\qquad\qquad\qquad\qquad\vdots
\end{aligned} \tag{8}$$

Soustava (8) vyjadřuje ideální stav, kdy první harmonická je rovna požadovanému výstupu, a všechny vyšší harmonické jsou nula. Tato soustava je předeterminovaná, proto vezmeme v úvahu jen prvních *s* rovnic. Všechny vyšší harmonické

$$V_k=\frac{4V_{DC}}{k\pi}\sum_{i=1}^{s}\cos(k\omega\theta_1),\qquad k=s+1,\ s+2,\ \ldots \tag{9}$$

nebudou nulové, ale jejich negativní vliv může být eliminován použitím vhodného filtru. Navíc jejich amplituda klesá podle vztahu (9) kde *k* je k-tá vyšší harmonická. Tudíž

$$\begin{aligned}
&\frac{4V_{DC}}{1\pi}\cdot\left(\cos(\omega\theta_1)+\cos(\omega\theta_2)+\ldots+\cos(\omega\theta_s)\right)=V_1\\
&\left(\cos(5\omega\theta_1)+\cos(5\omega\theta_2)+\ldots+\cos(5\omega\theta_s)\right)=0\\
&\left(\cos(7\omega\theta_1)+\cos(7\omega\theta_2)+\ldots+\cos(7\omega\theta_s)\right)=0\\
&\qquad\qquad\qquad\qquad\vdots\\
&\left(\cos(h_m\omega\theta_1)+\cos(h_m\omega\theta_2)+\ldots+\cos(h_m\omega\theta_s)\right)=0
\end{aligned} \tag{10}$$

kde

$$h_m=\begin{cases}3s-1 & :\ s\ sudé\\ 3s-2 & :\ s\ liché\end{cases}$$

a $\theta_1,\ldots,\ \theta_s$ jsou neznámé. Zde je potřeba aby pro neznámé opět platila nerovnost (2). Posloupnost lichých harmonických h_n = {1, 5, 7, 11,…, h_m} nedělitelných třemi je následující

$$h_n=6\left\lfloor\frac{n-1}{2}\right\rfloor+\begin{cases}1: & n\ liché\\ 5: & n\ sudé\end{cases},n=1,2,3,\ldots,s\,. \tag{11}$$

Principiální problém je řešení soustavy rovnic (10) Pokud je soustava rovnic v tomto tvaru, pak je pro výpočet nutné použít nějakou numerickou iterační metodu, například Newtonovu metodu.

Další možné úpravy mohou být provedeny za pomoci trigonometrické rovnice

$$\cos(n\omega\theta_i) = T_n(\cos(\omega\theta_i)) = T_n(x_i), \tag{12}$$

kde $T_n(x_i)$ je Chebyshův polynom prvního typu a stupně n. Soustava rovnic (10) může být upravena do tvaru

$$\sum_{i=1}^{s} x_i = m,$$

$$\sum_{i=1}^{s} T_j(x_i) = 0, \quad j = 5,7,11,...,h_m$$

což v rozepsané formě vypadá následovně

$$\begin{aligned}
p_1(x) &= \sum_{i=1}^{s} x_i - m = 0 \\
p_5(x) &= \sum_{i=1}^{s} \left(5x_i - 20x_i^{\,3} + 16x_i^{\,5}\right) = 0 \\
p_7(x) &= \sum_{i=1}^{s} \left(-7x_i + 56x_i^{\,3} - 112x_i^{\,5} + 64x_i^{\,7}\right) = 0 \\
&\vdots \\
p_{h_m}(x) &= \sum_{i=1}^{s} T_{h_m}(x_i) = 0.
\end{aligned} \tag{13}$$

Nyní je soustav rovnic (13) polynomiální v proměnných $x_1, x_2, \ldots, x_s$ pro které musí platit

$$0 < x_s < \cdots < x_2 < x_1 < 1. \tag{14}$$

Symbol m se nazývá modulační index a platí pro něj $m = \frac{V_1\pi}{4V_{DC}} = \frac{\pi}{4}\frac{V_1}{V_{DC}}$. Modulační index je číslo, které je rovno poměru V_1 k V_{DC}, kde V_1 je amplituda požadovaného sinusového signálu ($V_1 \sin(\omega t)$) a V_{DC} je napětí konkrétního stejnosměrného zdroje. Je zajímavé, že mohou být nalezena taková řešení, které generují na výstupu amplitudu V_1 která je jiná než součet napětí všech stejnosměrných zdrojů. V tomto případě lze použít vhodného spínání k řízení amplitudy výstupního signálu V_1 bez nutnosti měnit napětí těchto zdrojů.

Úhly mohou být vypočítány pomocí rovnice (12) s použitím vztahu

$$\theta_i = \frac{1}{\omega}\arccos x_i, \qquad i = 1,2,...,s. \tag{15}$$

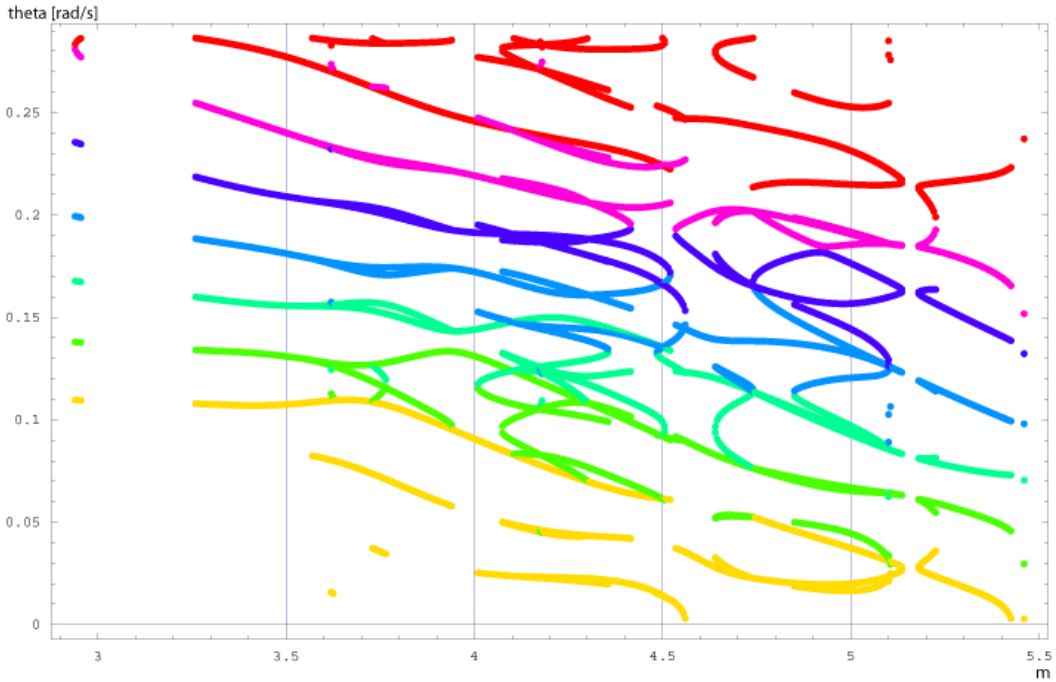
Původní problém (řešení transcendentální soustavy rovnic (10)) byl upraven na problém řešit soustavu polynomiálních rovnic (13). Tato polynomiální soustava může být vyřešena použitím více způsobů.

Avšak, jak je zřejmé, pro velké s (s = 6 je přiměřené) je daná soustava rovnic je poměrně velká, a tak její výpočet je velice složitý a nemůže být proveden bez znalosti a užití nějaké zvláštní struktury.

Po nalezení všech řešení pro dané m (předpokládá se různý počet řešení pro různé modulační indexy) se vybere to, které minimalizuje činitel harmonického zkreslení (total harmonic distortion – THD). Hodnota THD je definována takto

$$THD\ [\%] = 100 \sqrt{\frac{1}{V_1^2} \sum_{i=s+1}^{s+N} V_{h_i}^2}\ , \tag{16}$$

Kde N = 10 je pevně stanoveno a index h_i je počítán podle (11).



Obrázek 4: Závislost spínacích úhlů na modulačním indexu pro sedmizdrojový vícehladinový konvertor

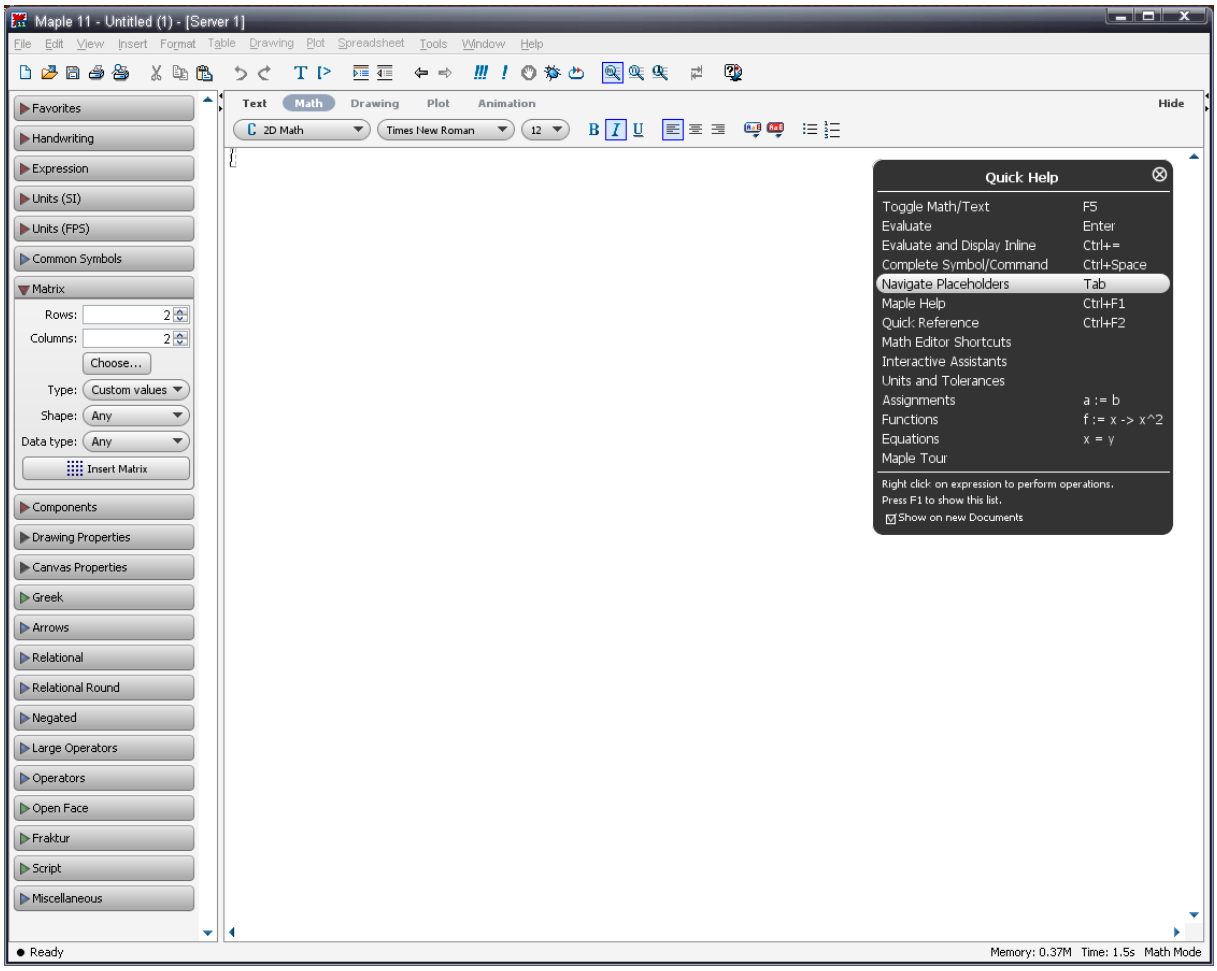
Pro toto řešení je výstupní signál po filtraci nejméně deformovaný nežádoucími vyššími harmonickými. Pro určitý modulační index m je možné, že soustava nebude mít žádné řešení. Pak je potřeba pracovat s jedním řešením, které minimalizuje čtverce vyšších harmonických (počínaje 5. harmonickou) a zaručuje rovnost první harmonické a požadovaného výstupu. Nebo, pokud je to možné, nejlepší varianta je vybrat jiný vhodný modulační index, pro který řešení soustavy (10) existuje. Další informace je možné získat v [1].

7. Přehled softwaru a jeho popis

7.1. Maple 11

7.1.1. Popis programu

Maple je systém počítačové algebry pro výuku a využití matematiky v přírodovědných, technických a ekonomických oborech, který byl vyvíjen od devadesátých let minulého století. Umožňuje symbolické a numerické matematické výpočty, jejich počítačovou vizualizaci, dokumentaci a publikaci. Učitelům, studentům i vědcům a výzkumným pracovníkům poskytuje uživatelsky přívětivé prostředí, ve kterém lze snadno používat matematiku.



Obrázek 5: Pracovní prostředí programu Maple 11

Současná verze 11 systému Maple (zkráceně Maple 11) umožňuje provádět jak symbolické a numerické výpočty a vytvářet grafy, tak doplňovat je vlastními texty a vytvářet tak tzv. hypertextové zápisníky. Takto vytvořené zápisníky umožňuje Maple 11 ukládat do souboru na počítači ve svém speciálním mapleovském formátu MW, který je uložen ve formátu XML. Soubory ve formátu MW umožňuje Maple 11 načítat zpět ke zpracování, což

umožňuje snadnou přenositelnost mapleovských zápisníků mezi nejrůznějšími počítačovými platformami a operačními systémy.

Samozřejmě, že tento produkt není freeware a je velice drahý. K dispozici je řada druhů licencí od nejjednodušší pro studenty, přes profesionální pro komerční organizace až k multilicencím pro akademické a státní instituce. Ceny se pohybují od $130 až k $1900 za licenci pro 1 instalaci. Katedra matematiky ČVUT disponuje multilicencí na více verzí tohoto systému. Avšak pro studenty a domácí použití je dostupná maximálně verze Maple 9. Ale z hlediska toho, že pro výpočet soustav polynomiálních rovnic se v současné verzi Maple 11 vyskytují nové algoritmy, konkrétně „Groebner basis algorithms of Maple“ (algoritmy počítající Göbnerovy báze), bylo potřeba využít verzi 11, ke které jsem měl přístup na několika fakultních počítačích.

Pro výpočet soustavy polynomiálních rovnic lze použít jednoduchého příkazu, který je velice efektivní:

```
solve(equations, variables);
```

kde *equations* je rovnice nebo nerovnice nebo soustava rovnic či nerovnic, *variables* jsou volitelně proměnné, které chceme vyřešit

Pokud Maple nedokáže vyřešit kořeny rovnice jedné proměnné, vrací výsledek s použitím výrazu *RootOf()* namísto skutečného výsledku. Použitím příkazu *fsolve* lze početně vypočítat řešení rovnice či jejich soustav.

```
fsolve(equations, complex);
```

kde *equations* je rovnice nebo soustava rovnic pomocí *complex* volitelně určujeme, zda chceme komplexní řešení.

Pro samostatnou polynomiální rovnici jedné proměnné příkaz *fsolve* vypočítá implicitně všechny reálné (nekomplexní) řešení. Pro obecnou rovnici, či soustavu rovnic vypočítá příkaz *fsolve* pouze jeden reálný kořen.

Nejvíce očekávané vlastnost programu Maple 11 je, že do něj byl integrován program J.C. Faugerea FGb. To je v podstatě knihovna napsaná v jazyku C, která je jedna z nejrychlejších pro výpočet Gröbnerových bází, kterou můžete v současné době získat. Podporuje výpočty s racionálními čísly koeficientů (pomocí modulárních algoritmů) a

s celými čísly délky p, kde p je méně než polovina délky slova použitého počítače. To je $p < 65536$ pro 32-bitové počítače nebo $p < 2^{32}$ pro 64-bitové počítače.

FGb počítá „celkový stupeň“ Gröbnerových bází (funkce *tdeg* v Maple) pomocí příkazu *Groebner[Basis]* automaticky, když je to možné. FGb také podporuje eliminační pořadí (*lexdeg* v Maple) ale implicitně se používá konverzní algoritmus Gröbnerových bází příkazu *Groebner[Basis]*.

Konverzní algoritmus Gröbnerových bází použije Gröbnerovy báze pro jedno pořadí neznámých pro vypočítání báze pro jiné pořadí. V podstatě příkaz *Groebner[Basis]* je požije pro výpočet lexikografických Gröbnerových bází, protože jsou často velmi těžké vypočítat přímo. Více informací lze získat na [4].

7.1.2. Instalace

Operační systém:	MS Windows, Linux, Macintosh, Solaris
Licence:	Placená
Velikost na disku:	650 MB

Tabulka 1: Shrnutí programu Maple 11

Instalaci programu Maple 11 jsem bohužel neprováděl. Zkusil jsem instalovat Maple 9, který mohu legálně užívat i doma díky multilicenci pro ČVUT. K objednání je spousta verzí tohoto programu v závislosti na preferovaném operačním systému. Osobně jsem opět zvolil Windows. Instalace proběhla velice jednoduše a na disku zabírá cca 650 MB, což je samozřejmě více, než programy které jsou určeny jen pro výpočet soustav polynomiálních rovnic, ale na druhou stranu je to více účelový software, který lze použít na řešení téměř všech matematických výpočtu z většiny vědních oboru. Také s narůstající diskovou kapacitou současných osobních počítačů bych velikost tohoto programu neviděl nikterak omezující.

7.1.3. Syntaxe

Opět jsem zvolil nejjednodušší soustavu rovnic o dvou neznámých. Nejprve si do proměnné *F* uložíme soustavu rovnic:

```
F := [rovnice_1, rovnice_2];
```

Následně pomocí příkazu solve() vyřešíme.

```
solve(F);
```

Vyzkoušíme i příkaz *fsolve()* , který by měl dát konkrétní řešení:

```
fsolve(F);
```

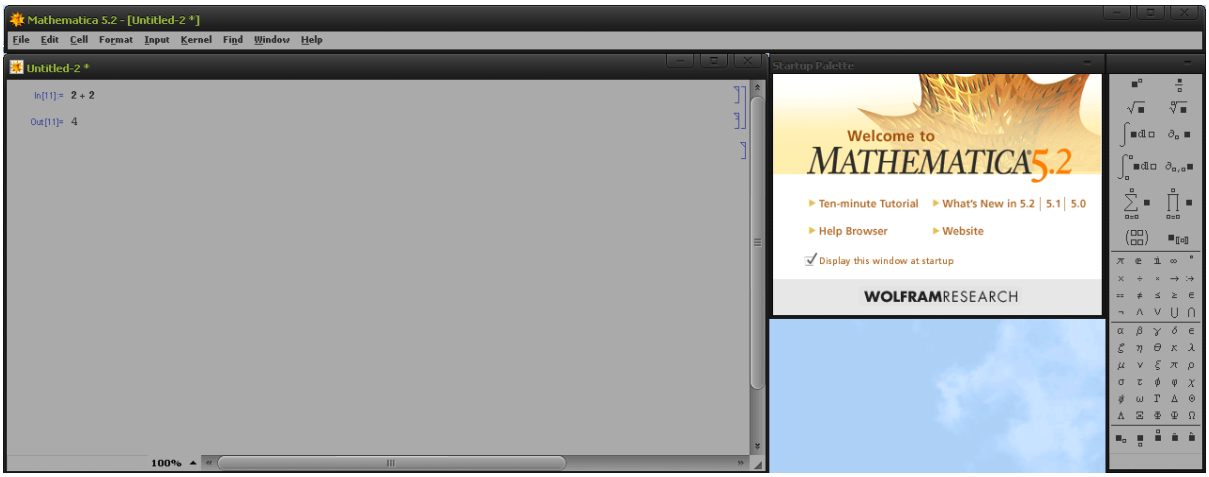
Toto je však pouze jedno možné řešení dané soustavy. Těch ale může a s největší pravděpodobností i bude existovat mnohem více. Výpočet pomocí obou příkazů proběhne celkem rychle a bez obtíží.

7.2. Mathematica 5.2

7.2.1. Popis programu

Tento software je obecné výpočetní prostředí spojující spoustu algoritmických, vizualizačních a uživatelských schopností do jednoho uživatelského prostředí, které se tváří jako běžný dokument. Původně byl tento software koncipován Stephenem Wolframem, následně vyvinut týmem matematiků a programátorů, které si najal, a nakonec je prodávaný jeho firmou Wolfram Research z Champaign z Illinois.

Od verze 1.0 z roku 1988 se Mathematica silně rozšířila a vyvinula do ještě více použitelného o a obecného výpočetního prostředí. Mimo to, že je vhodný téměř pro všechny odvětví matematiky, poskytuje multiplatformní podporu pro širokou škálu problémů, jako například interaktivní výpočetní prezentace, grafické a symbolické výpočty.



Obrázek 6: Pracovní prostředí programu Mathematica

Většina výukových a výzkumných organizací vlastní licenční servery, ale samozřejmě, že jsou v prodeji i individuální licence. Fakulta elektrotechnická ČVUT vlastní také tuto multilicenci, proto jsem mohl tento software plně vyzkoušet. Firma Wolfram Research také nabízí 15-ti denní trial verzi, kterou je možné objednat ke stažení zdarma. Ceny plné licence se pohybují od cca $50 pro studenty. Doplňující informace o tomto programu hledejte v [5].

7.2.2. Instalace

Operační systém:	MS Windows, Linux, Macintosh, Solaris, …
Licence:	Placená
Velikost na disku:	cca 300 MB (v závislosti na OS)

Tabulka 2: Shrnutí programu Mathematica 5.2

Osobně jsem vyzkoušel verzi Mathematica 5.2 pod operačním systémem Windows Vista. Instalace proběhala velmi jednoduše, tak jak jsme u Windowsovských aplikací zvyklí. Na disku tato aplikace zabírá cca 450 MB, což je srovnatelné s programem Maple. Jak už bylo zmíněno, na takto obsáhlý a sofistikovaný software je to stále dobré.

Uživatelské rozhraní se poněkud liší od posledních testovaných aplikací (Maple, MuPAD), avšak základní prvky má totožné. Zpočátku jsem měl problémy se syntaxí, poněvadž je také trochu odlišná (zejména typ použitých závorek), ale po několika vypočtených příkladech bylo vše v pořádku.

Pro řešení lineárních rovnic se používají Gaussovy eliminační a jiné metody. Pro samostatnou polynomiální rovnici se používá explicitní vzorec až do stupně čtyři, který se pokouší zjednodušit polynomy s použitím dalších funkcí (*Factor* a *Decompress*) nebo rozeznat nějaké speciální polynomy. Pro soustavu polynomiálních rovnic se vytvoří Gröbnerovy báze. Tyto funkce používají Buchbergerův algoritmus. Funkce *Reduce* pro soustavy polynomiálních rovnic využívá cylindrickou algebraickou dekompozici pro reálné domény a metody Gröbnerových bází pro komplexní domény. Cylindrická dekompozice využívá algoritmu Collins-Hong s projekcí Brown-McCallum nebo Hong.

7.2.3. Syntaxe

Pro řešení soustav rovni se tedy používá opět velice jednoduchý příkaz *Solve*, jehož kód včetně přidružených funkcí je cca 500 stránek dlouhý.

```
Solve[{eq1,eq2 , ... }, {var1,var2 , ... }];
```

Kde *{eq1,eq2 , ... }* je soustava rovnic a *{var1,var2 , ... }* jsou neznámé, pro které chceme soustavu vyřešit.

Také jsem zde poprvé narazil na problém, že je potřeba rovnice zadat ve formátu:

```
eq == 0;
```

To znamená, že vždy musí být rovnice ve tvaru rovnice. Například levá strana rovnice == (se rovná) pravá strana rovnice.

Opět tento příkaz nalezne obecná symbolická řešení dané soustavy, kdybychom chtěli početně vyčíslit všechny výsledky, můžeme rovnou použít příkaz *NSolve*:

```
NSolve[{eq1,eq2 , ... }, {var1,var2 , ... }];
```

Celkově na mě tento software působil velice dobře, zejména jeho intuitivnost a rozsáhlá dokumentace. Časově bych tento software zařadil mezi jeden z nejrychlejších.

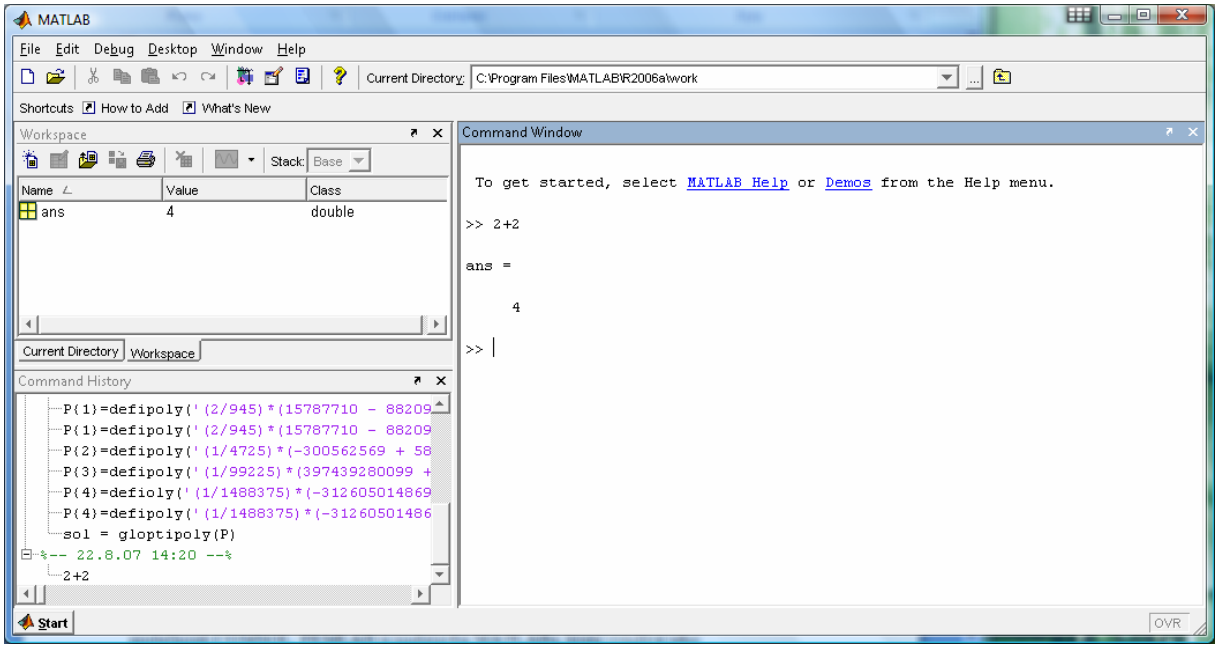
7.3. MATLAB

7.3.1. Popis

MATLAB je integrované prostředí pro vědeckotechnické výpočty, modelování, návrhy algoritmů, simulace, analýzu a prezentaci dat, měření a zpracování signálů, návrhy řídicích a komunikačních systémů. MATLAB je nástroj, jak pro pohodlnou interaktivní práci, tak pro vývoj širokého spektra aplikací.

Program existuje řadu let a prošel dlouhým vývojem. Vlastní Matlab není jen v jedné linii základního programu, ale používá se spousta rozšíření (toolbox). Nejznámější a asi nejpoužívanější je Simulink.

Simulink je program pro simulaci a modelování dynamických systémů, který využívá algoritmy Matlabu pro numerické řešení nelineárních diferenciálních rovnic. Poskytuje uživateli možnost rychle a snadno vytvářet modely dynamických soustav ve formě blokových schémat a rovnic.



Obrázek 7: Pracovní prostředí programu Matlab

V poslední době se také rozšířila nabídka výrobce o produkt FEMLAB švédské společnosti COMSOL. FEMLAB je nadstavba MATLABu, která využívá jeho matematických funkcí a grafického prostředí k modelování a k simulaci úloh z technické praxe v oblasti strojírenství, chemie, elektromagnetismu a z dalších oblastí fyziky. FEMLAB usnadňuje pochopení řady fyzikálních a technických procesů díky názorné grafice, snadné změně vstupních parametrů a možnosti simulace dané úlohy. Na řešení je možné v krátké době pohledět z několika zorných úhlů a efektivně tak najít optimální variantu.

Výhoda Matlabu je nejen v jeho velkých možnostech, ale i v tom, jak je široce rozšířen v průmyslu a jeho verze existují pro řadu operačních systémů (Unix, Linux, Windows, Open VMS, IRIX, Solaris, Macintosh, HP-UX a další). Matlab je komerční software, ale existuje i jeho GNU (general public license) varianta Octave. Není to samozřejmě to samé, ale syntaxe příkazů a práce s tímto produktem je stejná jako v Matlabu, chybí další nadstavby a vylepšení, ale pro samotné výpočty je to velmi zajímavý nástroj. Hlavně je zdarma a tak přístupnější studentům i pro domácí použití. Studenti mají možnost také výhodně si pořídit studentskou verzi Matlabu, kde je rovněž velmi výhodná cena proti komerční verzi. Další informace lze získat na [6].

7.3.2. Historie

Zkratka pro „**mat**rix **lab**oratory“, MATLAB, byla vynalezena v pozdních sedmdesátých letech 20. Století Cleverem Molerem, vedoucím katedry počítačových věd na univerzitě v Novém Mexicu. Původně tento program navrhl pro své studenty, aby měli přístup l LINPACKu a EISPACKu bez nutných znalostí Fortranu. Brzy se ale rozšířil na další univerzity a nalezl spoustu příznivců z řad lidí, kteří se zabývali aplikovanou matematikou. Když v roce 1983 navštívil Moler univerzitu v Stanfordu, kde pracoval inženýr Jack Little, tak mu představil tento program. Ten si uvědomil jeho komerční potenciál a přidal se k Molerovi a jeho kolegům. Společně přepsali MATLAB do jazyka C a v roce 1984 založili společnost MathWorks, aby mohli pokračovat ve vývoji. Tyto přepsané knihovny byly známé jako JACKPACK.

MATLAB si prvně osvojili inženýři, kteří se zabývali řízením, ale později se rozšířil do spousty jiných odvětví. Nyní je také hojně využíván k výuce lineární algebry a početní analýzy a v neposlední řadě také v odvětví zabývajícím se zpracováním obrazů.

Katedra řízení Fakulty elektrotechnické ČVUT disponuje multilicencí pro své zaměstnance i studenty tudíž jsem tento program opět mohl testovat doma. Vzhledem k tomu, že jsem již tento software používal dříve v rámci určitých předmětů během mého studia, bylo pro mě jeho použití vskutku jednoduché.

7.3.3. Instalace

Operační systém:	MS Windows, Linux, Macintosh, Solaris
Licence:	Placená
Velikost na disku:	500 MB (čistě MATLAB)

Tabulka 3 : Shrnutí programu MATLAB

Jak už zde bylo uvedeno, tento produkt lze nainstalovat téměř kamkoliv, proto jsem zvolil opět operační systém MS Windows. Verzi programu MATLAB jsem zkoušel R2006a. Instalace probíhala velice jednoduše a po skončení zabírala zhruba 1,6 GB, což je sice hodně, ale vzhledem k velice rozsáhlým možnostem využití tohoto programu je to samozřejmě potřeba.

7.3.4. Syntaxe

Pro výpočet soustavy polynomiálních rovnic lze použít jednoduchý příkaz:

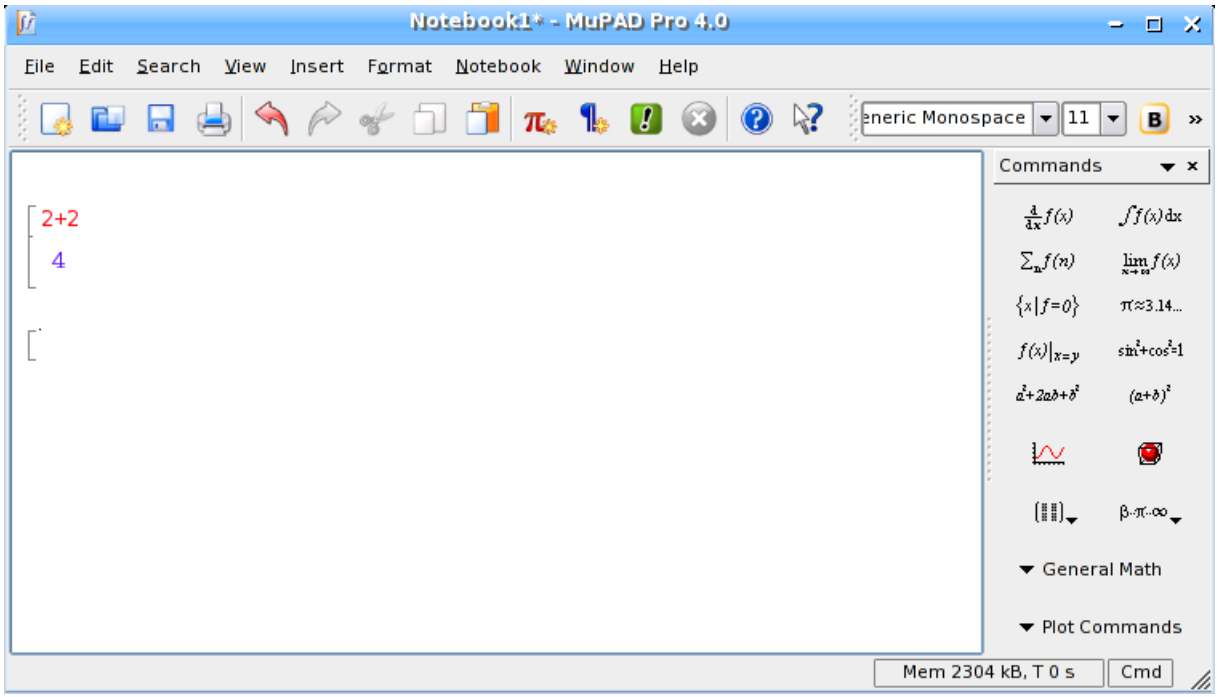
```
g = solve(eq1,eq2,...,eqn,var1,var2,...,varn);
```

kde *eq1* až *eqn* jsou jednotlivé polynomy a *var1* až *varn* jsou neznámé, pro které chceme soustavu řešit. Pokud nezadáme žádné neznámé, soustava bude vyřešena pro všechny. Vstupem *eq1* až *eqn* mohou být symbolické výrazy nebo řetězce definující daný polynom. Po provedení tohoto příkazu je na výstupu struktura, která obsahuje, v našem případě dvě (dvě neznámé), uspořádané matice výsledků. Ty si posléze musíme vyvolat příkazy (v našem případě proměnné x a y:

```
g.x
g.y
```

7.4. MuPAD

7.4.1. Popis programu



Obrázek 8 : Pracovní prostředí programu MuPAD

MuPAD (viz [7]) je počítačový algebraický software původně vyvinutý výzkumnou skupinou na Paderbornské univerzitě. Od roku 1997 je vyvíjen společností SciFace Software GmbH & Co. KG ve spolupráci s výzkumnou skupinou MuPADu a partnery z některých dalších univerzit. Od podzimu roku 2005 byla nabízena verze MuPAD Light zdarma pro výzkum a výuku, ale poněvadž se zavřel domovský institut výzkumné skupiny MuPADu, jsou dostupné pouze Pro verze, které už jsou placené. Naštěstí bylo možné stáhnout 30-denní trial verzi, kterou jsem mohl otestovat.

Z programátorského hlediska je hlavní jádro MuPADu napsáno v jazyce C, ostatní (knihovny) je napsáno v programovacím jazyce MuPAD.

7.4.2. Instalace

Operační systém:	MS Windows, Linux, Macintosh
Licence:	Placená, 30 denní trial
Velikost na disku:	200 MB

Tabulka 4 : Shrnutí programu MuPAD

Zřejmě z důvodu komerčního prodeje byla tato aplikace k dispozici pro téměř všechny operační systémy, což je velice pozitivní. Tentokrát jsem zvolil verzi pro Linux. Jelikož jsem zvolil verzi rpm instalace byla velice jednoduchá a následné spouštění též. Instalace na disku zabírá cca 150 MB což je na takto obsáhlý software velmi dobré.

Tento software je velice sofistikovaný a poskytuje velmi mnoho funkcí a matematických operací, od jednoduchých přímých výpočtu až po složité časově náročné operace. Grafické rozhranní tohoto programu je velice intuitivní. Dalo by se přirovnat ke grafickému rozhraní aplikace Maple. I pravidla psaní příkazů, zejména typy používaných závorek jsou velice podobné. Jako základní počítací algoritmus je zde opět použit algoritmus pro výpočet Gröbnerovych bází.

7.4.3. Syntaxe

Pro řešení polynomiálních rovnic, a jejich soustav se využívá opět standartního příkazu solve():

```
solve(system, variables);
```

kde *system* je rovnice nebo soustava rovnic a *variables* jsou proměnné, pro které chceme danou rovnici či soustavu řešit.

Následně jsem opět použil tento program k otestování na soustavě dvou rovnic o dvou neznámých, která se jevila jako nejjednodušší. Velice zajímavě na mě působilo to, že zpočátku mi dával podivné výsledky a chybová hlášení, ale když jsem zkoušel to samé po několikáté, tak už to byly výsledky příjemnějšího charakteru.

Nejdříve jsem si tedy do proměnné *G* vložil danou soustavu:

```
G=[eq1, eq2];
```

kde *eq1* a *eq2* jsou dané dvě rovnice, a dále jsem použil pouze již zmíněnou funkci *solve()*:

```
solve(G,[x,y]);
```

kde druhý parametr funkce jsou proměnné, pro které chceme soustavu vyřešit. Výsledkem tohoto příkazu ale stále nejsou početní hodnoty, nýbrž jen obecné řešení:

Abychom zjistili konkrétní hodnoty, musíme se pokusit výsledek vyčíslit například pomocí příkazu *float():*

```
float(%);
```

Symbol *%* zastupuje poslední vypočítanou hodnotu (výsledek příkazu), pokud bychom chtěli rovnou číselné hodnoty, museli bychom použít příkaz *solve()* uvnitř příkazu *float():*

```
float(solve(G,[x,y]));
```

Dále se v tomto programu nabízí použít příkaz *numeric::fsolve(),* který ale vrátí pouze jeden číselný kořen.

```
Numeric::fsolve(G,[x,y]));
```

Na druhou stranu lze u tohoto příkazu použít další nepovinný parametr *MultiSolutions*:

```
Numeric::fsolve(G,[x,y]), MultiSolutions);
```

který by měl zaručit to, že dostaneme všechna řešení. To se mi bohužel ale nepovedlo, stále tento příkaz vracel jen jedno možné řešení, stejné jako v případě bez tohoto parametru.

Rychlost výpočtů této jednoduché soustavy nepatřila mezi nejrychlejší, zejména při prvních, již zmíněných, neúspěšných pokusech, avšak stále byla celkem přijatelná.

7.5. Gloptipoly 2

7.5.1. Popis programu

GloptiPoly (viz [8]) je knihovna do programu Matlab, která umí vytvořit a vyřešit konvexní lineární maticové nerovnosti (linear matrix inequality – LMI) obecných

optimalizačních problémů s polynomiálními kritérii a omezením. Tyto algoritmy jsou založené na teorii, která je popsaná v [2]. GloptiPoly není zaměřená na obecné řešení nekonvexních optimalizačních problémů, ale umožňuje řešit řadu konvexních vztahů s rostoucími velikostmi, jejichž optima budou zaručeně monotónicky konvergovat ke globálnímu optimu.

GloptiPoly řeší LMI vztahy s pomocí další knihovny do programu Matlab, výpočetního zařízení SeDuMi (viz [9]), využívaje při tom výhod řídkosti a struktury speciálního problému. Volitelně lze společně s GloptiPoly použít uživatelsky příjemné rozhraní nazvané DefiPoly, založené na Symbolic Math Toolboxu Matlabu, které umožňuje zadávat dané vztahy symbolicky se syntaxí podobnou z programu Maple. GloptiPoly je zaměřená na řešení malých a středně velkých výpočetních problémů. Například dosud nedokáže pracovat s kvadratickými optimalizačními problémy s více než 19 proměnnými. Na zrušení tohoto omezení se však už v současné době pracuje a mělo by být odstraněno v dalších verzích. Početní experimenty v manuálu této knihovny ilustrují, že pro většinu příkladů je globální optimum dosaženo přesně pomocí LMI vztahů střední velikosti, při relativně malých systémových požadavcích.

Pracovní prostředí této aplikace se nijak neliší od pracovního prostředí aplikace Matlab (viz Obrázek 7), neboť je to pouze jeho knihovna.

7.5.2. Instalace

Operační systém:	Závislé na MATLABu
Licence:	freeware
Velikost na disku:	9 MB (včetně SeDuMi)

Tabulka 5: Shrnutí balíku GloptiPoly

Jak již zde bylo uvedeno, je to pouze knihovna funkcí do Matlabu, tudíž je závislá pouze na tomto softwaru, který je multiplatformní. Pro správný chod je potřeba mít verzi Matlabu alespoň 5.3 nebo vyšší. Osobně jsem tuto knihovnu zkoušel ve verzi 11 pro MS Windows. Dále je potřeba mít nainstalovanou knihovnu SeDuMi, jejíž instalace je také velmi jednoduchá. Stačí pouze přidat složku, kde je umístěné SeDuMi, a složku, kde je umístěná GloptiPoly, do pracovního adresáře Matlabu.

7.5.3. Syntaxe

Při samotném výpočtu je tedy zapotřebí nejprve definovat soustavu rovnic příkazem:

```
P{n} = defipoly('expr','vars');
```

Kde *n* je číslo rovnice (počítáno od čísla 1), *expr* udává konkrétní polynomiální výraz spolu s klíčovým slovem (*min*, *max*, ==, <=, >=) a *vars* jsou proměnné použité ve výrazu (oddělené čárkami). Několik těchto příkazů (pro *n* rovnic *n* příkazů) ve výsledku vytvoří speciální strukturu, se kterou se dále počítá pomocí příkazu:

```
out = gloptipoly(P);
```

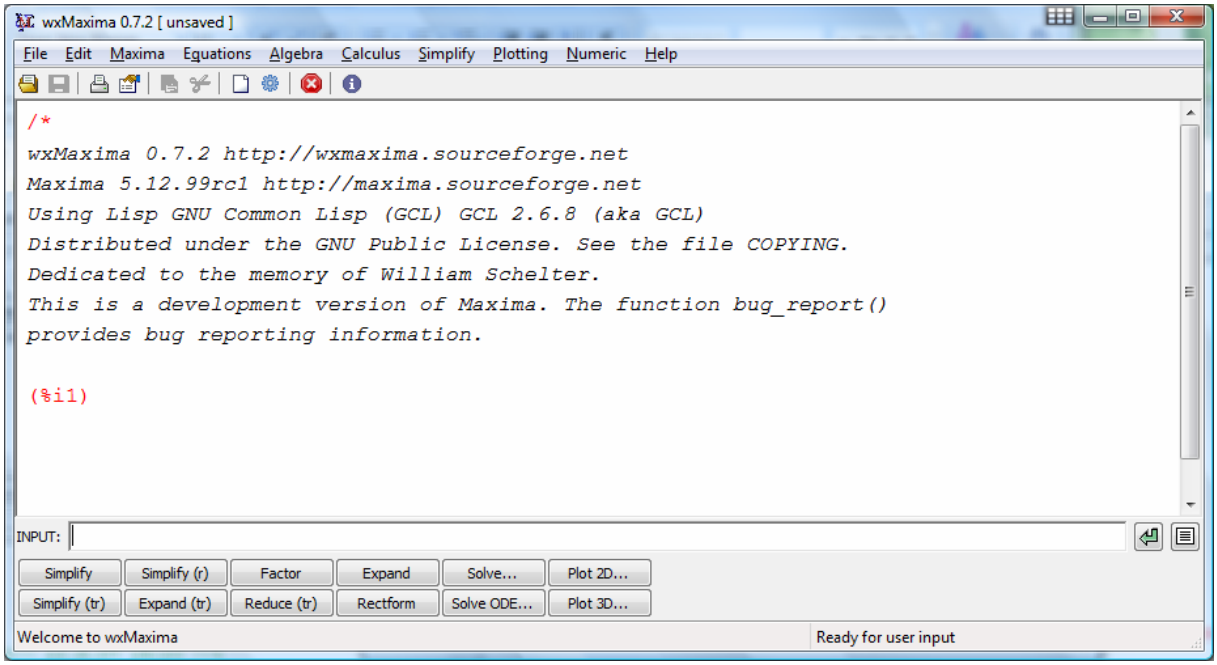
který provede výpočetní operace a výsledek uloží do struktury *out*, která obsahuje *status* (číselné označení statusu výsledku – 1 pro úspěšný výpočet, 0 pro případy, kdy nebylo možné zjistit globální optimum, a -1 pro případy, kdy LMI problém nemůže být řešen), dále obsahuje *sol* (což jsou konkrétní řešení) a *crit* (což je LMI kritérium). K těmto údajům přistupujeme pomocí tečky. Například pro zobrazení řešení soustavy:

```
out.sol
```

7.6. Maxima

7.6.1. Popis programu

Maxima je výpočetní systém pro manipulaci se symbolickými a numerickými výrazy, včetně diferenciálního a integrálního počtu, Taylorových řad, Laplaceovy transformace, obyčejných diferenciálních rovnic, soustav lineárních rovnic, polynomů a množin, seznamů, vektorů, matic a tenzorů. Tento software dosahuje velice přesných numerických řešení díky použití exaktních frakcí, libovolně přesných celých čísel a libovolně přesných čísel s pohyblivou desetinnou čárkou. Dále tato aplikace umí zobrazit grafy funkcí a dat ve dvou či třech dimenzích.



Obrázek 9: Pracovní prostředí programu Maxima

Zdrojový kód tohoto výpočetního systému může být spouštěn z mnoha operačních systému včetně Windows, Linux a MacOS X. Všechny tyto zdrojové kódy a předkompilované binární soubory jsou dostupné na webových stránkách SourceForge.net (viz [10]).

7.6.2. Historie

Maxima je potomek aplikace Macsyma, legendárního algebraického výpočetního systému vyvinutého v pozdních šedesátých letech minulého století v Technologickém institutu v Massachusetts. Je to jediný software založený na úspěchu systému stále veřejně dostupného a s velkou uživatelskou komunitou, díky tomu, že má povahu produktů Open Source. Ve svých dobách byla Macsyma velice revoluční a mnoho pozdějších výpočetních systému, jako například Maple a Mathematica, v ní hledalo inspiraci.

Odvětví Maximy od Macsymy bylo udržováno Williamem Schelterem od roku 1982 do doby kdy zemřel, v roce 2001. V roce 1998 dosáhl povolení vydávat zdrojový kód pod GNU General Public License (GPL). Byly to jeho úspěchy a zkušenosti, které zajistily, že software Maxima přežil dodnes. Jen díky jeho dobrovolnému úsilí a expertním znalostem se udržel původní kód Macsymy ve skvělém stavu. Po jeho smrti vznikla skupina uživatelů a vývojářů, která zajistila rozšíření Maximy k široké veřejnosti.

7.6.3. Instalace

Operační systém:	MS Windows, Linux
Licence:	freeware
Velikost na disku:	cca 25 MB (závislé na OS)

Tabulka 6 : Shrnutí programu Maxima

Osobně jsem vyzkoušel verzi pro MS Windows. Instalace probíhala opět velmi intuitivně a na disku zabírá zhruba 85 MB což je velice dobré pro takto rozsáhlý výpočetní systém. Uživatelské rozhraní tohoto softwaru je poněkud jednodušší než v předešlých případech (Mathematica, Maple), ovšem základní výpočetní funkce se objevily hned po spuštění aplikace v panelu nástrojů pod příkazovou řádkou. Dále také Maxima disponuje rozsáhlou nápovědou a dokumentací, až nebývalého obsahu, poněvadž nikde jinde jsem nenašel alespoň nástin algoritmu výpočtu funkce *solve()* v základní nápovědě aplikace.

7.6.4. Syntaxe

Při výpočtu je tedy využita následující metoda:

Nechť *E* je výraz a *X* je proměnná. Pokud je E lineární v proměnné X, pak je to triviálně řešeno pro X. Pokud je E ve tvaru A*X^N + B, pak výsledek je (-B/A)^1/N) krát N-tý kořen.

Pokud *E* není lineární v *X*, pak největší společný dělitel exponentů *X* v *E* (řekněme *N*) je rozdělen do exponentů a násobnost kořenů je násobena *N*. Následně je příkaz *solve* zavolán znovu na výsledek. Pokud *E* faktorizuje, pak je příkaz *solve* zavolán na každý z faktorů. Nakonec, pokud je to zapotřebí, musí funkce *solve* použít kvadratická, kubická a další pravidla.

V případě, že je E polynomiální v nějaké funkci nějaké proměnné, pro kterou chceme výraz řešit, řekněme F(x), pak je to nejprve řešeno pro F(x) (nazvěme výsledek C), pak rovnice F(x)=C může být řešena pro X poskytující inverzní funkci k funkci F.

Pro řešení soustav lineárních a nelineárních polynomiálních rovnic příkaz solve zavolá příkazy *linsolve* nebo *algsys* a vrátí seznam řešení pro dané proměnné.

Zvolil jsem si opět stejnou soustavu dvou rovnic o dvou neznámých jako v předešlých případech. Nejprve jsem si do proměnné *sys* uložil danou soustavu zde je opět potřeba dát pozor na to, že je zapotřebí psát výrazy ve tvaru *eq = O*:

```
sys:[eqn_1=0, eqn_2=0];
```

Do proměnné *var* jsem si uložil proměnné, které chci z dané soustavy vyřešit.

```
var:[x, y];
```

Jak už jsem tedy zmínil, opět se použije funkce:

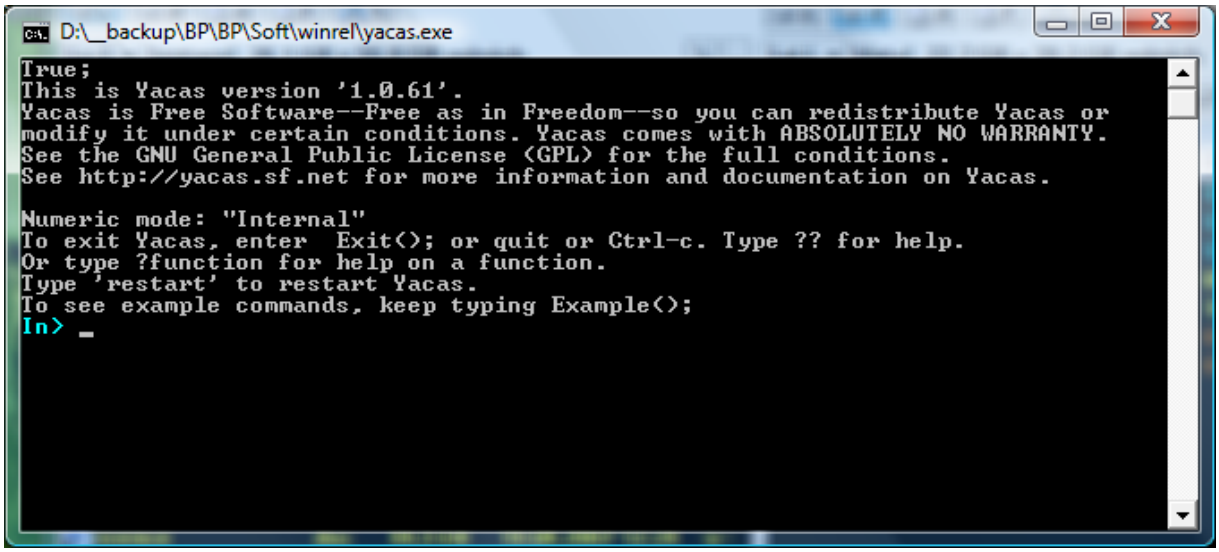
```
solve (sys, var);
```

Uvedený postup vyřeší soustavu početně a zřejmě pro všechny řešení a to i komplexní. Výsledky byly velmi podobné softwaru Mathematica. Časově byl na tom tento software poněkud hůře, ale stále výpočet netrval déle než pět vteřin, což můžeme považovat za úspěch.

7.7. Yacas

7.7.1. Popis programu

YACAS (Yet Another Computer Algebra System) je velice jednoduchý a obecný výpočetní algebraický systém, program pro symbolické výpočty matematických výrazů. Používá svůj vlastní programovací jazyk navržený jak pro symbolické, tak pro velice přesné numerické výpočty. Tento systém má knihovnu skriptů, které implementují spoustu matematických operací, a také podporu pro jednoduché vkládání nových skriptů. Dále má tento software velice obsáhlou dokumentaci (více než 320 stránek) svého skriptovacího jazyka funkcí, které jsou implementovány, a algoritmů, které jsou použity pro výpočet. Tento program je samozřejmě také Open Source, což znamená, že má svůj kód otevřený pro kohokoliv a každý může tento kód upravit podle svého.



Obrázek 10: Pracovní prostředí programu Yacas

Zdrojový kód aplikace YACAS se skládá přibližně z 22000 řádků napsaných v programovacím jazyku C++ a 13000 napsných ve skriptovacím jazyku se 170 funkcemi definovanými v jádře C++ a 600 funkcemi definovanými ve skriptovacím jazyce.

Více informací o tomto software lze najít v [11].

7.7.2. Instalace

Operační systém:	MS Windows. Linux, (web applet)
Licence:	freeware
Velikost na disku:	cca 1 MB

Tabulka 7 : Shrnutí programu Yacas

Mezi hlavní výhody tohoto softwaru patří to, že je zdarma (GPL), dálé jeho flexibilní a jednoduchý programovací jazyk s příjemnou a nastavitelnou syntaxí, multiplatformní využití (Unix, Windows, MacOS, EPOC32, atd.), malé systémové požadavky a jeho snadná rozšiřitelnost. Tento výpočetní systém je dokonce přístupný i z obyčejného webového prohlížeče, protože lze spustit interaktivně jako Java Applet.

Osobně jsem zkoušel nejprvne interaktivní Applet ale ten měl jistá omezení, zejména to, že do něj nelze vložit text ze schránky, což v případě velice komplikovaných soustav znamenalo takřka nemožnost použití. Následné jsem stáhnul překompilovanou verzi pro platformu Windows, kde vkládaní rovnic ze schránky bylo částečně implementováno, ale také podporovala načítaní příkazů ze souboru. Instalace nebyla nutná, program byl konsolového charakteru. Překvapila mě i velice malá velikost, zhruba 600 kB, což je nejlepší zatím ze všech testovaných programů.

7.7.3. Syntaxe

Pro výpočet rovnic jsou zde k dipozici dvě funkce, Solve() a OldSolve(). Autoři tohoto balíku ponechali záměrně dva různé příkazy ze dvou důvodů. První, filozofický, je ten, že je dobré mít více různých algoritmů pro tentýž výpočet. Druhý, více praktický důvod je ten, že nová verze (Solve) nedokáže pracovat se soustavami rovnic, vždy pouze s jednou, ale stará verze tohoto algoritmu (OldSolve) umí pracovat i se soustavami.

Příkaz:

```
OldSolve(eqlist, varlist);
```

se tedy pokouší vypočítat jednu nebo více rovnic. Při počítání soustav rovnic se tedy vypočítá soustava uložená v proměnné *eqlist* pro proměnné uložené v proměnné *varlist*. Výsledek tohoto výpočtu je seznam všech hodnot, které jsou ve stejném pořadí jako proměnné v seznamu *varlist*. Tento příkaz ale bohužel vrátí pouze jedno možné řešení.

Při řešení samostané rovnice se používá další funkce *SuchThat*. Soustava rovnic se řeší rekurzivně. Nejprve se vybere rovnice, ve které se jedna z proměnných vysktuje přesně jednou, pak je tato rovnice vyřešená pomocí *SuchThat* a následně je řešení substituováno do ostatních rovnic pomocí *Eliminate*, snižující počet rovnic o jednu. Tento potup je účinný pro všechny lineární rovnice a velkou skupinu rovnic nelineárních.

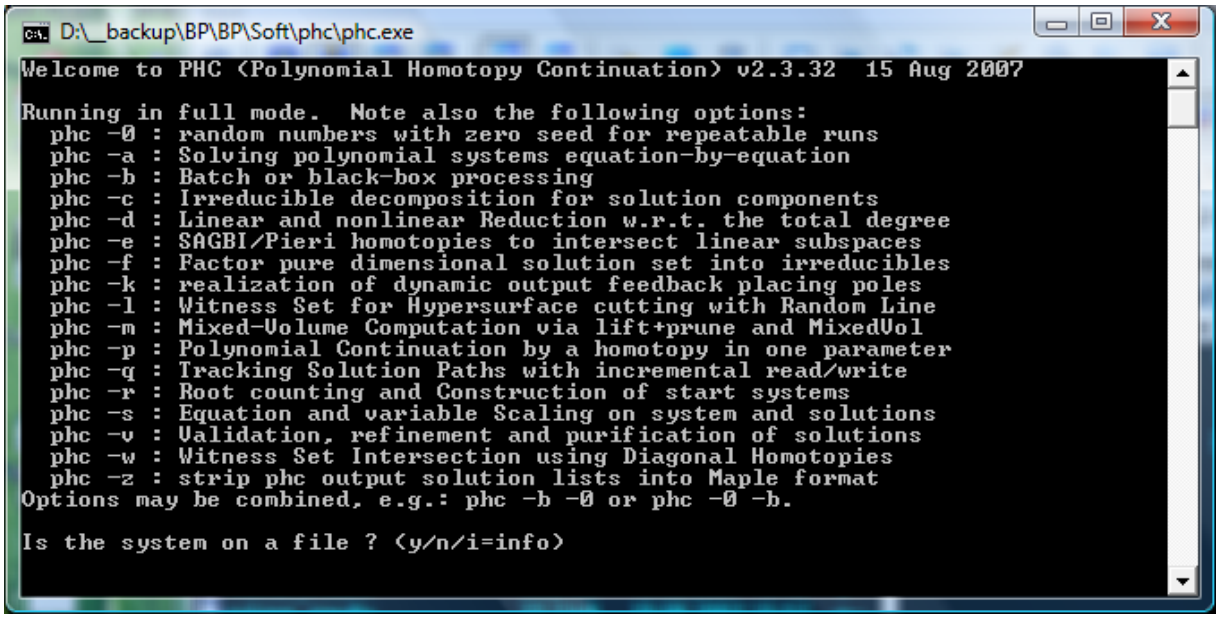
7.8. PHCpack

7.8.1. Popis

Tento software PHCpack (viz [12]) implementuje „Homotopy continuation“ metody k numerickému výpočtu aproximací ke všem izolovaným řešením systému *n* polynomiálních rovnic o *n* neznámých.

Název „*Polynomial Homotopy Continuation“* spojuje tři klíčové koncepty této metody. Poté, co vyřešíme soustavu *polynomiálních* rovnic, využijeme algebraickou strukturu ke spočítání kořenů a k vytvoření počátečního systému. Pomocí „*continuation*“ metod jsou známé řešení počátečního systému rozšířeny tak, aby obsahovaly požadované řešení cílové soustavy. Tato deformace je definována jako „*homotopy*“, to je skupina soustav spojujících počáteční a cílové soustavy.

Následující intuitivní úvaha může ulehčit tento složitý problém. Samozřejmě, že vyhodnocování polynomu o více neznámých může být učiněno v polynomiálním čase, tj. v čase úměrném polynomiální funkci v dimenzi, stupních a počtu členů. Spočítání jednoho řešení je těžké NP, protože můžeme použít následující nedeterministický algoritmus: Hádejte kořen a následně ověřte, zda se hodí k rovnici. Toto ověřování probíhá v polynomiálním čase. Vypočítání všech řešení je mnohem těžší, protože neexistuje algoritmus v polynomiálním čase k ověření toho, zda odhadnutý počet řešení je správný počet řešení. Tento výpočetní problém se označuje jako #P těžký. Z toho důvodu je náš problém velice nepoddajný pro zvětšující se dimenzi a zvyšující se počet stupňů. Následkem toho jsou výpočty omezeny na určitou dimenzi n a komplexnost je měřena v závislosti na velikosti výstupu, počtu řešení.



Obrázek 11: Pracovní prostředí programu PHCpack

Hlavní myšlenka v polynomiálních „*homotopy continuation*“ metodách je počítání kořenů, protože to určuje počet cest k řešení, které musí být sledovány. Nedávné výzkumy usilovaly o vyvinutí ostrých kořenů, které by vedly k homotopiím s optimálním počtem cest. Počet kořenů je také podstatný nástroj k ověřování početních řešení.

7.8.2. Instalace

Operační systém:	MS Windows, Linux, SunOS, Macintosh,…
Licence:	freeware
Velikost na disku:	cca 9 MB

Tabulka 8 : Shrnutí progamu PHCpack

Tento software je možné nalézt velmi snadno, je velice dobře zdokumentovaný a to nejen na jednom místě, existuje spoustu webových stránek zabývajících se tímto programem. Tudíž má i spoustu pomocných materiálů a tutoriálů, jak ho používat. Program je freeware, tudíž si ho můžete bezplatně stáhnou a nainstalovat na téměř libovolný počítač i libovolný operační systém, protože si lze jednoduše vybrat, pro jaký operační systém chcete balík stáhnout. Já osobně preferuji Windows, tudíž jsem program testoval na tomto OS. Velikost rozbaleného programu je cca 9 MB, v závislosti na vybraném OS. Instalace není potřeba, stejně tak není potřeba stahovat nějaké další knihovny či podpůrné programy pro správný chod aplikace.

7.8.3. Syntaxe

Po spuštění programu můžete zvolit, zda máte připravený nějaký soubor vstupních dat, nebo je zadáte postupně přímo do příkazové řádky programu.

Formát vstupního souboru je následující:

```
3 8
x11*x22 - x21*x12;
x12*x23 - x22*x13;
x13*x24 - x23*x14;
```

kde první číslo na prvním řádku (3) označuje počet rovnic a druhé číslo na prvním řádku (8) označuje počet neznámých dané soustavy. Následuje n řádků rovnic ukončených středníkem, kde n je počet rovnic.

Následuje spousta možných nastavení, metod a podmínek k dosažený spravných výsledků.

Výsledek procedury tento program ukládá do speciálního výstupního souboru, ze kterého je pak možné řešení analyzovat. Komplexní hodnoty jsou vypsány jako dvojice reálných a imaginárních čísel

Zvolil jsem soustavu ze zadání pro pětizdrojový konvertor o dvou rovnicích a dvou neznámých. Při spouštění programu jsem použil parametr „-b“, a proto jsem nemusel dále už nic nastavovat. Výsledky i rychlost byly celkem uspokojivé.

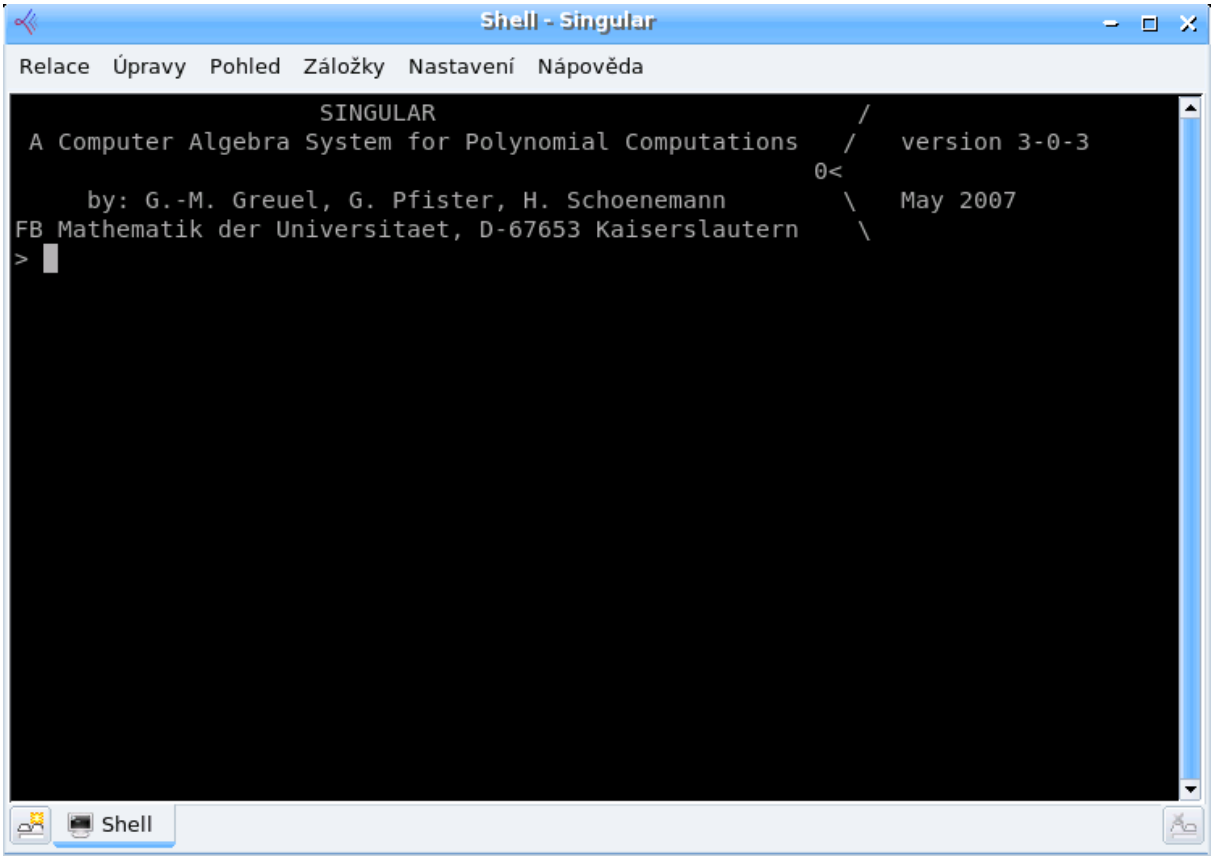
7.9. SINGULAR

7.9.1. Popis programu

Singular je další z počítačových výpočetních systémů zejména určených pro polynomiální výpočty s důrazem kladeným na komutativní algebru, algebraickou geometrii a singulární teorii.

Tento balík obsahuje jeden z nejrychlejších a nejobecnějších druhů algoritmů pro výpočet Gröbnerových, respektive standartních bází. Jeho implementace obsahuje Buchbergův algoritmus a Morův algoritmus pro vyjímečné případy. Dále poskytuje řadu výpočetních nástrojů, například k faktorizaci (rozklad na činitele) polynomů, výpočtu výslednic, charakteristických množin, nejvetších společných dělitelů a ke spoustě jiných výpočetních aplikací.

Je založen na velice uživatelsky jednoduché interaktivní příkazové řádce a programovacím jazyku typu C, a jeho funkčnost je rozšířena díky knihovnám napsaným v programovacím jazyku Singular. Tyto knihovny jsou stále možné uživatelsky doplňovat. Díky velice obecné a efektivní implementaci komunikačních způsobů je tento balík schopný nabídnout své funkce přístupné i jiným programům. Další informace lze získat na [13].



Obrázek 12 : Pracovní prosředí programu Singular

7.9.2. Historie

Vývoj tohoto softwaru začal v roce 1984 implementací algoritmu „Mora's Tangent Cone“ do Modula-2 na počítačích Atari (K.P. Neuendorf, G. Pfister, H. Schönemann; Humboldt-Universitätzu Berlin). Potřeba nového systému vyrostla z výzkumů matematických problémů vycházejících ze singulární teorie, které žádný dosavadní systém nebyl schopný řešit.

V počátku devadesátých let minulého století se domov projektu Singular přestěhoval do Kaiserslauternu, byl napsán a implementován základní algoritmus, postavený na obecných bazích, v jazyce C a nakonec byl portován na různé operační systémy (Unix, MS-DOS, Windows NT a MacOS.

Potřeba dalších rozšíření (jako například faktorizace polynomů, výpočet největších společných dělitelů, atd.) a upřesnění vedla v roce 1997 k vydání programu Singular verze 1.0 a v roce 1998 k vydání verze 1.2 (kde byly mnohem rychlejší algoritmy k počítání standartních a Gröbnerových bází založené na Hilbertových množinách a vylepšených implementací vlastního algoritmu, knihoven pro primární dekompozici, prstencových normalizací, atd.)

7.9.3. Instalace

Operační systém:	MS Windows, Linux, Macintosh
Licence:	freeware
Velikost na disku:	cca 50 MB

Tabulka 9: Shrnutí programu Singular

Jak už jsem zde uvedl, tento software má multiplatformní podporu, proto jsem ho nejprve zkoušel na MS Windows Vista. Konkrétně v tomto případě jsem příliš neuspěl, poněvadž pro běh tohoto programu bylo zapotřebí emulátorů Cygwin, jehož instalace z neznámého důvodu na tento operační systém nebyla možná. Proto jsem tedy zkusil distribuci pro Linux, konkrétně jsem použil soubor RPM, díky kterému byla instalace velice jednoduchá. Na oficiálních stránkách tohoto softwaru uvádějí, že je zapotřebí alespoň 20 MB volného místa na disku, což je velice příjemné, ve skutečnosti je to ale více.

7.9.4. Syntaxe

Samotné použití tohoto balíku už není tak jednoduché jako v případě některých jiných sofistikovanějších programů, ale na internetu existuje velice obsáhlá dokumentace se spousty příkladů, ze kterých jsem čerpal. Ve srovnání s jinými programy jsou tyto manuály a tutoriály velice užitečné a dostačující, což se o některých říct nedá.

Nyní už konkrétní jednoduchý příklad. Opět jsem zvolil soustavu dvou rovnic o dvou neznámých.

Nejprve je potřeba načíst speciální knihovnu pro řešení rovnic:

```
LIB "solve.lib";
```

Následně vytvořit prstenec:

```
ring r=0,x(1..2),dp;
```

Následně je potřeba definovat rovnice (v proměnných *x(1)* a *x(2)*) a ideály:

```
ideal i=f0,f1,f2,f3,f4;
ideal si=std(i);
```

f0 až *fn* jsou polynomy.

Po těchto krocích už můžeme aplikovat samotný výpočetní proces *solve()*:

```
def T=solve(i,30,1,"nodisplay");
```

který vypočítá všechny řešení včetně jejich násobností, vytvoří prstenec, ve kterém je uložen seznam SOL všech možných (i komplexních) řešení. Pro přístup k těmto řešením stačí napsat:

```
setring T;
SOL[1][1];
```

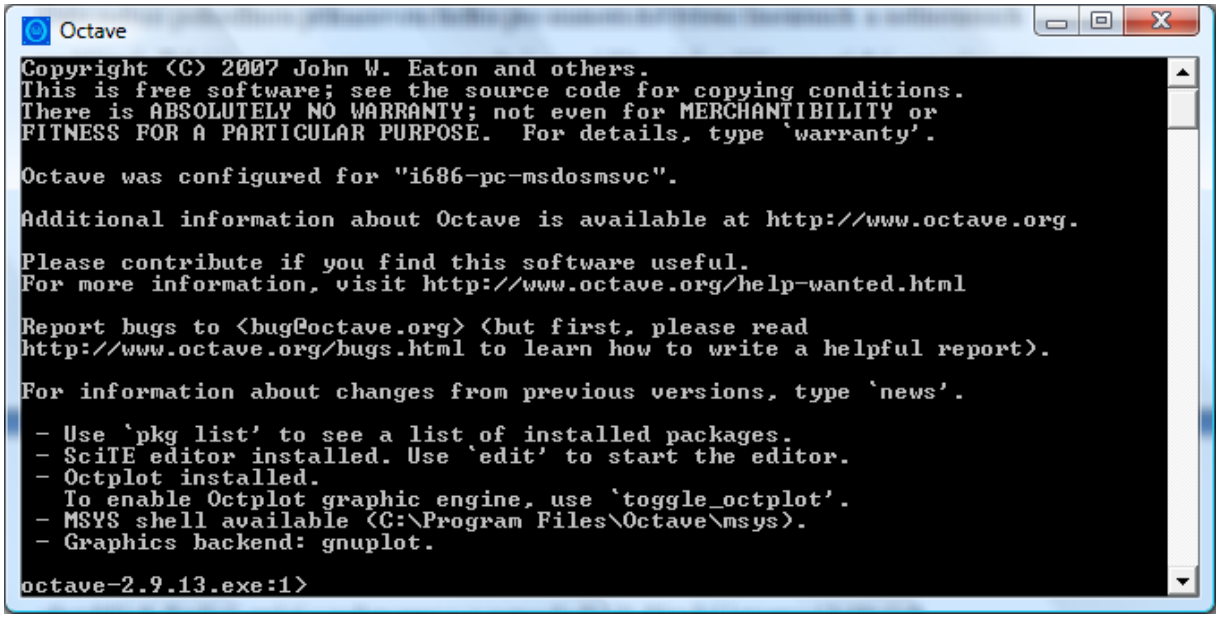
Ke zjištění konkrétních číselných řešení stačí použít příkaz SOL, ale ten vrátí celý seznam, který vznikl po příkazu *solve()*, tj. obsahuje i násobnosti řešení.

7.10. OCTAVE

7.10.1. Popis programu

Octave (viz [14]) je jazyk vysokého stupně, který byl primárně vynalezen pro numerické výpočty. Dále nabízí pohodlnou příkazovou řádku pro numerické řešení lineárních a nelineárních problémů. Také je v tomto programu možné provádět nejrůznější numerické experimenty s použitím jazyka, který je velice podobný a kompatibilní s jazykem Matlabu.

Tento software má také spoustu doplňkových nástrojů pro řešení základních numerických problémů z lineární algebry, pro nalezení kořenů nelineárních rovnic, integraci obvyklých funkcí, práci s polynomy a v neposlední řadě také integraci běžných diferenciálních a diferenčně-algebraických rovnic. Tento program je také jednoduše rozšiřitelný a upravitelný pomocí uživatelsky definovaných funkcí napsaných ve vlastním programovacím jazyku Octave, nebo přímo dynamicky načítaných modulů napsaných v jazyce C++, C, Fortran a mnohých dalších.



Obrázek 13: Pracovní prostředí programu Octave

Veliká výhoda této aplikace je její možnost volné distribuce. Tento produkt spadá pod licenci GNU GPL (General Public Licence), proto jí můžete distribuovat a upravovat při dodržení podmínek této licence.

Octave byl původně napsán Johnem E. Eatonem, ke kterému se připojila spousta dalších. Poněvadž je toto software typu Open Source, může každý pomoci s jejím vývojem napsáním dalších rozšiřujících funkcí a podáváním zpráv o všech vzniklých problémech při užívání.

7.10.2. Historie

Octave byl původně koncipovaný (asi v roce 1988) jako doprovodný software k výukové knize o návrhu chemických reaktorů, jejímiž autory jsou James B. Rawlings z Wisconsinko-Madisonské univerzity John G. Ekerdt z univerzity v Texasu. Původně si představovali nějaký velice specializovaný nástroj pro řešení problému plynoucích z návrhů chemických reaktorů. Později, když viděli omezení takového přístupu, tak se rozhodli postavit mnohem více flexibilní nástroj.

Stále ale byli nějací lidé, kteří tvrdili, že by měli požívat programovací jazyk Fortran, protože to je jazyk inženýrů, ale pokaždé když se o to pokoušeli, studenti strávili spoustu času zjišťováním, proč jejich kód napsaný ve Fortranu spadl, nemajíce dostatek času na skutečné učení chemického inženýrství. Věřili, že s právě takovým interaktivním prostředím, které

nabízí Octave, bude většina studentů schopná pochytit základy velice rychle a začít s ním pracovat jistě během pár hodin.

Vývoj na plný úvazek začal na jaře roku 1992. První testovací verze vyšla 4. ledna roku 1993 a verze 1.0 byla vydána 17 února roku 1994. Od té doby prošel tento software několika důležitými změnami, byl implementován do několika distribucí Linuxu (Debian, SuSE) a popsán v červencovém vydání měsíčníku Linux Journal.

Samozřejmě, že je Octave v poslední době mnohem více než další pomůcka pro výuku bez dalšího možného využití za zdmi vyučovací třídy. I když jejich původní představy byly poněkud nejasné, věděli, že chtějí vytvořit něco, co umožní studentům řešit reálné problémy, něco co budou moci použít i na jiné věci než k návrhu chemických reaktorů. V dnešní době používá tento software tisíce lidí po celém světě k výuce, výzkumu ale i ke komerčnímu využití.

I když si všichni myslí, že název Octave má něco společného s hudbou, tak je to vlastně jméno jednoho dřívějšího profesora jednoho z autorů, který napsal slavnou knihu zabývající se chemickými reakcemi a který byl také známý díky své schopnosti velice rychle počítat (v originále napsané – „back of the envelope calculations“ – výpočty na zadní straně obálky).

7.10.3. Instalace

Operační systém:	MS Windows, Linux, Macintosh, Solaris
Licence:	freeware
Velikost na disku:	cca 80 MB (v závislosti na OS)

Tabulka 10 : Shrnutí programu Octave

Jelikož je tento balík k dispozici pro různé operační systémy, opět jsem zvolil instalaci pro MS Windows. Instalace byla velice jednoduchá a na disku zabírá cca 80 MB.

Problém tohoto programu, alespoň pod MS Windows, je ten, že do příkazové řádky nelze vložit obsah schránky, což pro takto komplikované rovnice je veliký problém. Proto jsem musel hledat alternativní řešení a to načtení příkazů z externího souboru.

7.10.4. Syntaxe

Pro řešení soustavy je nejprve nutné vytvořit funkci, která bude obsahovat všechny rovnice:

```
function y = f (x)
y(1) = …
y(2) = …
…
endfunction
```

Následně už stačí jen danou funkci vyřešit pomocí příkazu:

```
[x, info] = fsolve ("f", [1; 1; …])
```

kde f je daná funkce a druhý argument jsou počáteční body.

8. Další nalezený software

Samozřejmě, že na internetu je k dispozici celá řada dalších programů pro výpočet soustav polynomiálních rovnic. Snažil jsem se zde zahrnout všechny, které se mi podařilo nainstalovat a pomocí kterých jsem vypočítal alespoň jednoduchou soustavu.

Problémy s instalací jsem měl zejména tehdy, pokud se jednalo o aplikaci určenou pro operační systém Linux, jelikož s tímto nemám příliš velké zkušenosti. Jednalo se zejména o knihovnu Synaps, kde bylo potřeba nainstalovat spoustu jiných podpůrných knihoven a balíků, zejména pak překladače Fortran. Toto se mi bohužel nepovedlo, i když následné použití vypadalo celkem jednoduše a bylo dobře zdokumentované.

Pokud se mi povedlo program nainstalovat, stále jsem neměl vyhráno. Bylo nutné pročíst dokumentaci a naučit se řešit soustavy polynomiálních rovnic. To byl další problém. Některé aplikace, například Macaulay 2, neměly danou problematiku dostatečně zdokumentovanou. Konkrétně v tomto případě sice on-line manuál měl přímo kapitolu „řešení soustav polynomiálních rovnic“, avšak po kliknutí na odkaz se na stránce objevil pouze název kapitoly a prázdná stránka.

Některé programy se mi podařily jednoduše nainstalovat a i když jsem po dlouhém hledání našel nějaký návod, jak počítat daný problém, tak se mi s ním nepodařilo vypočítat ani jednodušší soustavu polynomiálních rovnic. Toto se stalo například u programu CoCoA.

9. Testování na konkrétní úloze

V této části práce si vyberu jednu ze složitějších soustav polynomiálních rovnic a pokusím se ji vypočítat ve všech zde uvedených programech a výsledky nějak zhodnotit. S největší pravděpodobností neuspěji u všech programů z důvodů jistých omezení a komplikací.

Jako testovanou soustavu jsem si vybral soustavu pro sedmi-zdrojový vícehodinový konvertor, tudíž soustavu čtyř rovnic o čtyřech neznámých. Tyto rovnice už byly upravené pomocí substituce za mocninné součty. Dle grafu závislosti spínacích úhlů na modulačním indexu (viz Obrázek 4) je vidět, že řešení závisí ještě na modulačním indexu, ten jsem pro mé výpočty zvolil 3. Výsledné rovnice, které jsem použil v tomto testu, jsou k nahlédnutí v příloze.

Problémy se vyskytly i v syntaxi, protože některé programy si nebyly schopné poradit s proměnnou typu „p2“, a pro řešení bylo proto nutné rovnice upravit do správných syntaxí (například „a, b, c, d“ nebo „x(1), x(2), x(3), x(4)“). Konkrétní zdrojové kódy pro jednotlivé programy jsou opět přiloženy k této práci.

9.1. Konfigurace použitého počítače

Téměř všechny tyto programy jsem testoval na jednom počítači, až na Mathematicu a Maple, které jsem musel testovat na školních počítačích, z důvodů omezené licence. Všechny ostatní jsem testoval na notebooku následující konfigurace:

Procesor:

Intel Core Duo T2080 s frekvencí 1,73 GHz

1 MB L2 cache

FSB 533 MHz

Operační paměť:

1 x 1 GB DDR2 533/667 MHz

Operační systém:

Microsoft Windows Vista Home Premium

Mandriva Linux 2007.0

9.2. Výsledky jednotlivých testů

9.2.1. Mathematica

Tento software prokázal asi nejlepší výsledky, a to jak přesné, tak velice rychlé. Danou soustavu vyřešil za necelé 2 sekundy a řešení, včetně komplexních, našel 27. Tyto výsledky by se daly považovat za nejpřesnější, protože je všeobecně známo, že tento software dokáže počítat velice přesně. Tomu odpovídá i jeho cena.

9.2.2. Matlab

Matlab si se svojí funkcí solve() nedokázal poradit s takto velkou rovnicí. Po několika pokusech a ověření správné syntaxe stále hlásil varování:

```
Warning: Explicit solution could not be found.
```

a žádné řešení nenalezl. Při použití funkce solve() bez druhého volitelného parametru, což je seznam proměnných v dané soustavě, počítal velice dlouho, přibližně 30 sekund, než sdělil tuto varovnou hlášku. To bylo způsobeno tím, že se nejprve snažil v dané soustavě použitím příkazu findsym najít všechny proměnné. Při použití příkazu solve() s oběma parametry tuto hlášku sdělil již po necelé 1 sekundě. Každopádně mě tento program velice zklamal.

9.2.3. Gloptipoly

Doufal jsem, že tento balík pomůže Matlabu vyřešit danou soustavu. Leč nestalo se tak. Po nadefinování všech polynomů a spuštění samotného počítacího algoritmu, se program pokoušel spočítat danou soustavu. Asi po 80 sekundách, po vytvoření LMI vztahů, jejich vypočítání a numerickém dosazení skončil program při extrahování samotných výsledků. Chybové hlášení bylo následující:

```
Incomplete basis
Impossible to detect global optimality
```

9.2.4. Octave

V tomto programu, jak už bylo uvedeno, byl největší problém to (alespoň pod MS Windows), že nebylo možné vložit rovnice ze schránky, což při takto komplikované soustavě působilo značně nepoužitelně. Avšak bylo možné vytvořit soubor, který následně pomocí

jednoduchého příkazu zpracujeme. I po prolomení této bariéry však tento software nedosáhl uspokojujících výsledků. Nejen, že mu bylo nutné zadat nějaký počáteční bod, ale výsledek vypočítal pouze jeden po cca 26 sekundách s varovnou hláškou:

```
iteration is not making good progress
```

Nikde v manuálu jsem se nedočetl jak získat všechna možná řešení. Zřejmě to ani nejde. Proto je tento produkt také nepoužitelný.

9.2.5. PHCpack

Použití tohoto programu se zdálo být velice jednoduché. Při spouštění z příkazové řádky stačilo zadat parametr „–b“ pro použití „black box“ výpočtů, při kterém není nutné zadávat další možnosti, jméno vstupního souboru a jméno výstupního.

Bohužel došlo k nečekaným komplikacím, a to, že tento program nezvládl přečíst rovnice použité v předchozích programech. Zdálo se, jako by nedokázal správně pracovat s proměnnými. Při zadání výrazu například ve tvaru „a / 11“ ho nedokázal správně interpretovat, bylo nutné výraz upravit do tvaru „(1 / 11) * a“. Pro správné načtení rovnic bylo nutné tyto rovnice upravit v jiném programu. Použil jsem Mathematicu a její funkci *Factor[]*.

Dále už tento program počítal dle očekávání. Výpočet probíhal necelých 5 minut a ve výsledku vrátil celkem 17 platných řešení, z toho 13 reálných a 4 komplexní. Vygenerovaný soubor je bohužel tak rozsáhlý, že nemůže být přiložen.

9.2.6. Singular

Při řešení soustavy rovnic tímto programem jsem se nesetkal s žádnými většími komplikacemi. Definování jednotlivých rovnic proběhlo celkem svižně, v řádu desítek až stovek milisekund. Samotný numerický výpočet konkrétních řešení trval přibližně 11 minut, což je poměrně dlouhý čas, ale výsledky byly uspokojivé. Vrátil, stejně jako Mathematica, 27 řešení.

9.2.7. MuPAD

Při opakovaných pokusech vypočítat danou soustavu tímto programem jsem nikdy nedospěl k pořádným výsledkům. V první řadě bylo zvláštní, že po definování rovnic a proměnných a následném zavolání funkce *solve()* nic nevypočítal a pouze přepsal příkaz. Po několika opětovných pokusech se dal do počítání. Ale i tak cca po 1 hodině a 15 minutách skončil neúspěšně, s varováním, že nemůže spočítat Gröbnerovy báze.

9.2.8. Maple

Tato aplikace poskytla celkem rychlé algoritmy k výpočtu dané soustavy, avšak numerické řešení našla pouze jedno, což považuji za velké mínus. Jedno konkrétní řešení vypočítala za necelé dvě sekundy, ale když to porovnám například s Mathematicou, která za 2 sekundy nalezla všechna možná, tak je to velice slabé.

9.2.9. Maxima

Tato aplikace mě velice zklamala. Po prvotních testech na jednoduché úloze jsem do ní vkládal velké naděje. Avšak ve finále to bylo horší. Po nadefinování všech rovnic a proměnných jsem zavolal funkci *solve()*. Čas k vykonání tohoto příkazu přerůstal přes normální meze. Po nastavení nejvyšší priority procesu Maxima a vypnutí všech ostatních nepotřebných procesů v systému jsem se stále nemohl dostat řešení, nebo alespoň varovnou hlášku. Po dvou různých pokusech, kdy jsem se pokusil rovnice nějak ještě upravit a kdy výpočet trval přes 4,5 hodiny, jsem se rozhodl proces ukončit. Závěr je tedy takový, že Maxima jednoduché soustavy počítat umí, ale i tak používá horší algoritmy než ostatní software. Se složitějšími soustavami si ale bohužel neporadí, respektive možná poradí, ale v nelidském čase.

9.2.10. Yacas

S tímto programem, jak už zde bylo uvedeno, byly problémy zejména v zadávání komplikovaných rovnic. Naštěstí tento program lze spouštět s nepovinnými parametry, jako například jméno vstupního souboru pro zpracování. Dále bylo nutné vyřešit to, abych viděl výsledek procesu, poněvadž pokud se načte soubor pro zpracování, výsledek standardně nevypíše. To se mi podařilo odstranit použitím speciální funkce *ToFile()*, také zde bylo nutné zvýšit počítací hloubku. Avšak i po vyřešení všech těchto prvotních problémů, tento program

takto komplikovanou soustavu vyřešit neuměl a vrátil prázdnou množinu. A to vše asi po 5,5 minutách.

9.3. Porovnání výsledků

Vzhledem k tomu, že rozumné řešení se mi podařilo nalézt jen ve třech případech, omezím se jen na tyto aplikace: Mathematica, PHCpack, Singular.

software	Mathematica	PHCpack	Singular
přibližná doba výpočtu	2 sec	5 min	11 min
počet nalezených řešení	27	17	27
reálné	17	13	17
komplexní	10	4	10

Tabulka 11 : Porovnání dosažených výsledků

Je velice zajímavé, že Mathematica dokáže vypočítat všechna řešení ve velmi krátkém čase. Oproti tomu program Singular vypočital také všechna řešení, ale zabralo mu to poněkud delší dobu. Což ale u takto složité soustavy je stále celkem dobré. Vzhledem k tomu, že ostatním programům se to nepodařilo vůbec, můžeme ho označit jako velice schopný.

Program PHCpack potřeboval poněkud kratší dobu k výpočtu, avšak nenašel všechna řešení. Tento nedostatek by se teoreticky mohl dát odstranit použitím různých nastavení v průběhu výpočtu. Já jsem však zvolil „black box solver“ a tomuto jsem se vyhnul.

Pokud bychom se tedy nevázali časově a potřebovali bychom přesný výpočet, je asi vhodné použít program Singular. Na druhou stranu, pokud bychom chtěli rychlejší proces bez zbytečného zdržování s nastavováním parametrů výpočtu, pak by bylo vhodné sáhnout po programu PHCpack.

Samozřejmě, že nikdy nemůžeme na sto procent tvrdit, že nalezená řešení jsou opravdu všechna, a to ani při použití programu Mathematica. Vždy existuje alespoň malé procento, kdy se může mýlit. Jelikož ale Singular nalezl stejný počet řešení, můžeme tvrdit, že jsou opravdu všechny.

Je zřejmé, že jasným vítězem je program Mathematica, který ale bohužel nespadá do kategorie freeware. Jediná polehčující okolnost je ta, že Fakulta matematiky ČVUT disponuje multilicencí, kterou lze použít.

V příloze najdeme výstupy programů Mathematica a Singular. Vygenerovaný soubor aplikace PHCpack je natolik dlouhý, že nemůže být přiložen, avšak na doprovodném CD je samozřejmě k nahlédnutí

10. Závěr

V této práci jsem se snažil sestavit ucelený přehled softwaru pro řešení soustav polynomiálních rovnic. Tato problematika je velice komplikovaná, speciálně pro tak složité rovnice, jako pro ty, které vedou k řešení spínacích úhlů vícehladinových konvertorů.

Na internetu je k dispozici celá řada nástrojů, které umí tyto soustavy počítat, ne všechny jsou ale jednoduché k použití a ne všechny dokážou dané konkrétní soustavy skutečně řešit. Pokusil jsem se vybrat ty, které byly dostatečně zdokumentovány a které dokázaly vypočítat alespoň jednu z jednodušších rovnic.

Nejprve jsem se zaměřil na některé známé výpočetní programy, které nejsou zcela zdarma, jako například Mathematica, Maple, atd. a v další části už skutečně pouze na freeware.

Některé z nalezených programů počítaly velice svižně, další z nich zase počítaly poměrně dlouho s využitím téměř všech systémových prostředků. Několikrát i počítač přestal zcela reagovat a bylo ho nutné restartovat. Tyto rozdíly jsou zřejmě způsobeny různými použitými algoritmy k výpočtu.

Při testování jsem se také musel naučit používanou syntaxi jednotlivých programů. Ta byla u všech téměř stejná, přesto však v jistých ohledech odlišná. Rozdíly byly zejména v přiřazování a závorkách. Tyto odchylky také nepatrně komplikovaly vytváření této práce.

Nejlepších výsledků jsem dosáhl při použití freewarů Singular a PHCpack. Tyto prokazovaly celkem dobré vlastnosti. Zejména to byly jediné programy, které soustavu dopočítaly do konce a současně našly, údajně, všechny řešení. Program Singular, který byl sice o něco pomalejší, podle mého mínění, zvládl výpočet ze všech freewarů nejlépe.

Pokud bych tedy měl doporučit nějaký program pro řešení soustav polynomiálních rovnic, asi bych volil nejprve Mathematicu (v případě dostupnosti nějaké licence), dále bych asi doporučil Singular a následně PHCpack, jehož použití bylo takřka nejjednodušší.

V příloze této práce jsou vytištěné konkrétní pracovní soubory programů Mathematica a Singular.

11. Použité zdroje

11.1. Literatura

[1] M. Šebek, M. Hromčík, P. Kujan: Numerical and Symbolical Polynomial Methods, Odborná rozprava, 2006.

[2] J. B. Lasserre. Global Optimization with Polynomials and the Problem of Moments. SIAM Journal on Optimization, Vol. 11, No. 3, pp. 796–817, 2001.

[3] Pavel Kočička, Filip Blažek: Praktická typografie, Computer Press, Brno, ISBN 80-722-6385-4, 2004.

11.2. Elektronické zdroje

[4] Math Software for Engineers, Educators & Students [online]. Maplesoft, 2007 [cit. 2007-05-04]. Dostupný z WWW: <http://www.maplesoft.com/>.

[5] Wolfram Research, Inc. [online]. c2007 [cit. 2007-05-10]. Dostupný z WWW: <http://www.wolfram.com/>.

[6] The MathWorks - MATLAB and Simulink for Technical Computing [online]. c1994 [cit. 2007-06-10]. Dostupný z WWW: <http://www.mathworks.com/>.

[7] SciFace Software - Mathematics mastered with MuPAD Pro [online]. c2004 [cit. 2007-06-15]. Dostupný z WWW: <http://www.mupad.de/>.

[8] GloptiPoly 2 [online]. [2006] , Last updated on 25 April 2007 [cit. 2007-06-27]. Dostupný z WWW: <http://www.laas.fr/~henrion/software/gloptipoly/>.

[9] SeDuMi - Home [online]. [2006] [cit. 2007-06-27]. Dostupný z WWW: <http://sedumi.mcmaster.ca/>.

[10] Maxima - A GPL CAS based on DOE-MACSYMA [online]. [2003] [cit. 2007-08-05]. Dostupný z WWW: <http://maxima.sourceforge.net/>.

[11] Yacas for EPOC [online]. 2004 , Last update : March 22nd 2004 [cit. 2007-08-06]. Dostupný z WWW: <http://www.robertocolistete.net/Yacas/>.

[12] Distribution of the second public version of PHCpack [online]. [2004] [cit. 2007-08-10]. Dostupný z WWW: <http://www.math.uic.edu/~jan/download.html>.

[13] SINGULAR [online]. [2005] [cit. 2007-08-11]. Dostupný z WWW: <http://www.singular.uni-kl.de/>.

[14] Octave [online]. c1998-2006 [cit. 2007-08-16]. Dostupný z WWW: <http://www.gnu.org/software/octave/>.

[15] Wikipedie : Otevřená encyklopedie [online]. 12.8.2007 Dostupný z WWW: <http://cs.wikipedia.org/>.

[16] Wikipedia : The Free Encyclopedia [online]. 15.8.2007. Dostupný z WWW: <http://en.wikipedia.org/ >.

[17] FARKAŠOVÁ, Blanka, KRČÁL, Martin. *CITACE :* projekt Bibliografické citace [online]. verze 1.1. Brno : BB Group, c2004-2005 [cit. 2005-03-10]. Dostupný z WWW: <http://www.citace.com/apl-www.php>.

12. Příloha A – seznam obrázků

Obrázek 1: Jednofázová struktura vícehladinového kaskádního H-můstkového invertoru 9
Obrázek 2: Výstupní průběh 11-hladinového kaskádního invertoru 10
Obrázek 3: Výstupní průběh jednoho stejnosměrného zdroje 11
Obrázek 4 : Závislost spínacích úhlů na modulačním indexu pro sedmizdrojový vícehladinový konvertor 14
Obrázek 5: Pracovní prostředí programu Maple 11 15
Obrázek 6: Pracovní prostředí programu Mathematica 18
Obrázek 7: Pracovní prostředí programu Matlab 21
Obrázek 8 : Pracovní prostředí programu MuPAD 23
Obrázek 9: Pracovní prostředí programu Maxima 28
Obrázek 10: Pracovní prostředí programu Yacas 30
Obrázek 11: Pracovní prostředí programu PHCpack 33
Obrázek 12: Pracovní prosředí programu Singular 35
Obrázek 13: Pracovní prostředí programu Octave 38

13. Příloha B – seznam tabulek

Tabulka 1: Shrnutí programu Maple 11 17
Tabulka 2: Shrnutí programu Mathematica 5.2 19
Tabulka 3: Shrnutí programu MATLAB 22
Tabulka 4: Shrnutí programu MuPAD 24
Tabulka 5: Shrnutí balíku GloptiPoly 26
Tabulka 6: Shrnutí programu Maxima 29
Tabulka 7: Shrnutí programu Yacas 31
Tabulka 8: Shrnutí progamu PHCpack 33
Tabulka 9: Shrnutí programu Singular 36
Tabulka 10: Shrnutí programu Octave 39
Tabulka 11: Porovnání dosažených výsledků 45

14. Příloha C – obsah přiloženého CD

- **freeware** - složka s programy
  - **gloptipoly** - složka Gloptipoly
    - Gloptipoly.ZIP - archiv funkcí Gloptipoly
    - SeDuMi_1_1.zip - archiv podpurné knihovny SeDuMi
  - **maxima** - složka Maxima
    - maxima-5.12.99rc2.exe - instalace Maximy pro Windows
    - maxima-5.12.99rc2-1.centos4.i386.rpm - instalace Maximy pro Linux
  - **octave** - složka Octave
    - octave-2.9.13-setup.exe - instalace Octave pro Windows
  - **phcpack** - složka PHCpack
    - aix_phcv23p.tar.gz - instalace PHCpack pro Aix
    - lin_phcv23p.tar.gz - instalace PHCpack pro Linux
    - mac_phcv23p.tar.gz - instalace PHCpack pro MacOS
    - phc.exe - instalace PHCpack pro Windows
    - sun_phcv23p.tar.gz - instalace PHCpack pro Sun
  - **singular** - složka Singular
    - Singular-3-0-3-Full.exe - instalace Singular pro Windows
    - Singular-core-3.0.3-2.i386.rpm - jádro Singularu pro Linux
    - Singular-core-static-3.0.3-2.i386.rpm - jádro Singularu pro Linux (statické)
    - Singular-help-3.0.3-2.noarch.rpm - nápověda Singularu
    - Singular-libs-3.0.3-2.noarch.rpm - knihovny Singularu
    - Singular-release-3.0.3-2.noarch.rpm - vydání Singularu
  - **yacas** - složka Yacas
    - yacas-1.1.7.tar.gz - zdrojové soubory Yacasu pro Linux
    - Yacas-Windows-Console-1.0.61.zip - archiv souborů Yacasu pro Windows
- bp_2007_stefap1.pdf - vlastní bakalářská práce
- **worksheets** - složka výsledků testování
  - gloptipoly.txt - výstup programu Gloptipoly
  - maple.mw - výstup programu Maple
  - mathematica.nb - výstup programu Mathematica
  - matlab.mat - výstup programu MATLAB
  - maxima.wxm - výstup programu Maxima
  - mupad.txt - výstup programu MuPAD
  - octave.txt - výstup programu Octave
  - phcpack.txt - výstup programu PHCpack
  - singular.txt - výstup programu Singular
  - yacas.txt - výstup programu Yacas

15. Příloha D – výstup programu Singular

```
 A Computer Algebra System for Polynomial Computations   /   version 3-0-3
                                                       0<
     by: G.-M. Greuel, G. Pfister, H. Schoenemann        \   May 2007
FB Mathematik der Universitaet, D-67653 Kaiserslautern    \
> LIB "solve.lib";
// ** loaded /usr/share/Singular/LIB/solve.lib (1.36,2007/05/07)
// ** loaded /usr/share/Singular/LIB/triang.lib (1.11,2006/12/06)
// ** loaded /usr/share/Singular/LIB/elim.lib (1.21,2006/08/03)
// ** loaded /usr/share/Singular/LIB/poly.lib (1.46,2007/07/25)
// ** loaded /usr/share/Singular/LIB/ring.lib (1.31,2006/12/15)
// ** loaded /usr/share/Singular/LIB/primdec.lib (1.135,2007/04/20)
// ** loaded /usr/share/Singular/LIB/absfact.lib (1.6,2007/07/13)
// ** loaded /usr/share/Singular/LIB/matrix.lib (1.37,2007/04/20)
// ** loaded /usr/share/Singular/LIB/random.lib (1.17,2006/07/20)
// ** loaded /usr/share/Singular/LIB/inout.lib (1.28,2006/07/20)
// ** loaded /usr/share/Singular/LIB/general.lib (1.54,2007/01/08)
> ring r=0,x(0..3),dp;
> poly f0= (2/945)*(15787710 - 882090*x(0) - 9098001*x(0)^2 + 4083156*x(0)^3 -
498960*x(0)^4 - 34763256*x(1) + 27667332*x(0)*x(1) - 6237000*x(0)^2*x(1) +
194040*x(0)^3*x(1) + 18480*x(0)^4*x(1) + 543312*x(1)^2 - 665280*x(0)*x(1)^2 +
221760*x(0)^2*x(1)^2 + 591360*x(1)^3 - 49280*x(0)*x(1)^3 + 13013946*x(2) -
13471920*x(0)*x(2) + 2993760*x(0)^2*x(2) + 997920*x(1)*x(2) + 498960*x(0)*x(1)*x(2)
- 110880*x(0)^2*x(1)*x(2) - 443520*x(1)^2*x(2) + 10611216*x(3) - 3991680*x(0)*x(3)
-     1774080*x(1)*x(3) + 147840*x(0)*x(1)*x(3));
> poly f1= (1/4725)*(-300562569 + 583821108*x(0) + 154864710*x(0)^2 -
273007800*x(0)^3 + 75724740*x(0)^4 - 5896800*x(0)^5 +     355210596*x(1) -
1425214440*x(0)*x(1) + 699950160*x(0)^2*x(1) - 89107200*x(0)^3*x(1) -
764400*x(0)^4*x(1) + 218400*x(0)^5*x(1) + 174975840*x(1)^2 -
44291520*x(0)*x(1)^2 - 36691200*x(0)^2*x(1)^2 + 5241600*x(0)^3*x(1)^2 -
873600*x(1)^3 + 15142400*x(0)*x(1)^3 - 582400*x(0)^2*x(1)^3 - 228943260*x(2) +
517233600*x(0)*x(2) - 258869520*x(0)^2*x(2) + 23587200*x(0)^3*x(2) +
179262720*x(1)*x(2) - 94348800*x(0)*x(1)*x(2) + 34070400*x(0)^2*x(1)*x(2) -
873600*x(0)^3*x(1)*x(2) +     41932800*x(1)^2*x(2) - 10483200*x(0)*x(1)^2*x(2) -
1164800*x(1)^3*x(2) - 130319280*x(2)^2 + 70761600*x(0)*x(2)^2 -
19656000*x(1)*x(2)^2 - 2620800*x(0)*x(1)*x(2)^2 -     187405920*x(3) +
345159360*x(0)*x(3) - 47174400*x(0)^2*x(3) + 2620800*x(1)*x(3) -
45427200*x(0)*x(1)*x(3) + 1747200*x(0)^2*x(1)*x(3) - 94348800*x(2)*x(3) +
3494400*x(1)*x(2)*x(3));
> poly f1= (1/4725)*(-300562569 + 583821108*x(0) + 154864710*x(0)^2 -
273007800*x(0)^3 + 75724740*x(0)^4 - 5896800*x(0)^5 +     355210596*x(1) -
1425214440*x(0)*x(1) + 699950160*x(0)^2*x(1) - 89107200*x(0)^3*x(1) -
764400*x(0)^4*x(1) + 218400*x(0)^5*x(1) + 174975840*x(1)^2 -
44291520*x(0)*x(1)^2 - 36691200*x(0)^2*x(1)^2 + 5241600*x(0)^3*x(1)^2 -
873600*x(1)^3 + 15142400*x(0)*x(1)^3 - 582400*x(0)^2*x(1)^3 - 228943260*x(2) +
517233600*x(0)*x(2) - 258869520*x(0)^2*x(2) + 23587200*x(0)^3*x(2) +
179262720*x(1)*x(2) - 94348800*x(0)*x(1)*x(2) + 34070400*x(0)^2*x(1)*x(2) -
873600*x(0)^3*x(1)*x(2) +     41932800*x(1)^2*x(2) - 10483200*x(0)*x(1)^2*x(2) -
1164800*x(1)^3*x(2) - 130319280*x(2)^2 + 70761600*x(0)*x(2)^2 -
19656000*x(1)*x(2)^2 - 2620800*x(0)*x(1)*x(2)^2 -     187405920*x(3) +
345159360*x(0)*x(3) - 47174400*x(0)^2*x(3) + 2620800*x(1)*x(3) -
45427200*x(0)*x(1)*x(3) + 1747200*x(0)^2*x(1)*x(3) - 94348800*x(2)*x(3) +
3494400*x(1)*x(2)*x(3));
// ** redefining f1 **
> poly f2 = (1/99225)*(397439280099 + 329991732324*x(0) - 421931821878*x(0)^2 -
3560261040*x(0)^3 + 98622970740*x(0)^4 -     32785709040*x(0)^5 + 3761394840*x(0)^6
- 107956800*x(0)^7 - 1636694532864*x(1) + 316557175032*x(0)*x(1) +
871137117144*x(0)^2*x(1) -     494439573600*x(0)^3*x(1) + 92959201440*x(0)^4*x(1) -
5229907200*x(0)^5*x(1) - 101959200*x(0)^6*x(1) + 3998400*x(0)^7*x(1) +
1332486106560*x(1)^2 -     1433315701440*x(0)*x(1)^2 + 494466819840*x(0)^2*x(1)^2 -
56780478720*x(0)^3*x(1)^2 - 143942400*x(0)^4*x(1)^2 + 191923200*x(0)^5*x(1)^2 +
291492956160*x(1)^3 -     132506976000*x(0)*x(1)^3 + 9484204800*x(0)^2*x(1)^3 +
511795200*x(0)^3*x(1)^3 - 10662400*x(0)^4*x(1)^3 + 14343060480*x(1)^4 +
480942801612*x(2) -     174127532544*x(0)*x(2) - 266556906000*x(0)^2*x(2) +
155667536640*x(0)^3*x(2) - 24511591440*x(0)^4*x(2) + 647740800*x(0)^5*x(2) -
670698121968*x(1)*x(2) +     871525992960*x(0)*x(1)*x(2) -
```

```
295161088320*x(0)^2*x(1)*x(2) + 25453814400*x(0)^3*x(1)*x(2) +
683726400*x(0)^4*x(1)*x(2) - 23990400*x(0)^5*x(1)*x(2) -
309735256320*x(1)^2*x(2) + 151888020480*x(0)*x(1)^2*x(2) -
8636544000*x(0)^2*x(1)^2*x(2) - 383846400*x(0)^3*x(1)^2*x(2) -
25941619200*x(1)^3*x(2) +     767692800*x(0)*x(1)^3*x(2) + 64739379600*x(2)^2 -
87565302720*x(0)*x(2)^2 + 33131941920*x(0)^2*x(2)^2 - 1295481600*x(0)^3*x(2)^2 +
95520176640*x(1)*x(2)^2 -     60311865600*x(0)*x(1)*x(2)^2 +
2806876800*x(0)^2*x(1)*x(2)^2 + 47980800*x(0)^3*x(1)*x(2)^2 +
15545779200*x(1)^2*x(2)^2 - 127948800*x(1)^3*x(2)^2 -     14315071680*x(2)^3 +
7772889600*x(0)*x(2)^3 - 2159136000*x(1)*x(2)^3 - 287884800*x(0)*x(1)*x(2)^3 +
570067424640*x(3) + 145448654400*x(0)*x(3) -     275740174080*x(0)^2*x(3) +
69725698560*x(0)^3*x(3) - 4318272000*x(0)^4*x(3) - 1115167011840*x(1)*x(3) +
589079473920*x(0)*x(1)*x(3) - 71635334400*x(0)^2*x(1)*x(3) -
191923200*x(0)^3*x(1)*x(3) + 159936000*x(0)^4*x(1)*x(3) - 82296668160*x(1)^2*x(3) -
4606156800*x(0)*x(1)^2*x(3) + 1535385600*x(0)^2*x(1)^2*x(3) +
4094361600*x(1)^3*x(3) - 341196800*x(0)*x(1)^3*x(3) + 312408472320*x(2)*x(3) -
224722874880*x(0)*x(2)*x(3) + 20727705600*x(0)^2*x(2)*x(3) +
84734092800*x(1)*x(2)*x(3) +     1151539200*x(0)*x(1)*x(2)*x(3) -
767692800*x(0)^2*x(1)*x(2)*x(3) - 3070771200*x(1)^2*x(2)*x(3) -
10363852800*x(2)^2*x(3) + 383846400*x(1)*x(2)^2*x(3) +     202555745280*x(3)^2 -
27636940800*x(0)*x(3)^2 - 12283084800*x(1)*x(3)^2 + 1023590400*x(0)*x(1)*x(3)^2);
> poly f3 = (1/1488375)*(-31260501486963 + 19188051308244*x(0) +
41732561289078*x(0)^2 - 28455463927800*x(0)^3 - 2271003773220*x(0)^4 +
5919328270800*x(0)^5 -     1703105724600*x(0)^6 + 174410560800*x(0)^7 -
3619728000*x(0)^8 + 104257636328700*x(1) - 161054917560360*x(0)*x(1) +
1606903265520*x(0)^2*x(1) +     66688737937920*x(0)^3*x(1) -
30842436819600*x(0)^4*x(1) + 5239234526400*x(0)^5*x(1) - 278920152000*x(0)^6*x(1) -
3954888000*x(0)^7*x(1) +     134064000*x(0)^8*x(1) - 62986064732640*x(1)^2 +
159790227750720*x(0)*x(1)^2 - 115727564937600*x(0)^2*x(1)^2 +
33037452806400*x(0)^3*x(1)^2 -     3491267875200*x(0)^4*x(1)^2 +
17696448000*x(0)^5*x(1)^2 + 8043840000*x(0)^6*x(1)^2 - 10964215866240*x(1)^3 +
19834148275200*x(0)*x(1)^3 -     6206251564800*x(0)^2*x(1)^3 +
111541248000*x(0)^3*x(1)^3 + 58273152000*x(0)^4*x(1)^3 - 1072512000*x(0)^5*x(1)^3 +
2210385945600*x(1)^4 -     933514444800*x(0)*x(1)^4 + 308883456000*x(0)^2*x(1)^4 -
17160192000*x(0)^3*x(1)^4 + 2745630720000*x(1)^5 - 45760512000*x(0)*x(1)^5 +
1906688000*x(0)^2*x(1)^5 -     26965270240956*x(2) + 65207817581760*x(0)*x(2) -
12895583505840*x(0)^2*x(2) - 24324003345600*x(0)^3*x(2) +
12543171958800*x(0)^4*x(2) -     1908320601600*x(0)^5*x(2) +
50676192000*x(0)^6*x(2) + 17687097610560*x(1)*x(2) - 1026035895225600*x(0)*x(1)*x(2)
+ 92464641763200*x(0)^2*x(1)*x(2) -     26591406710400*x(0)^3*x(1)*x(2) +
2116334304000*x(0)^4*x(1)*x(2) + 40621392000*x(0)^5*x(1)*x(2) -
1876896000*x(0)^6*x(1)*x(2) + 14997601785600*x(1)^2*x(2) -
24906486528000*x(0)*x(1)^2*x(2) + 7958575296000*x(0)^2*x(1)^2*x(2) +
90091008000*x(0)^3*x(1)^2*x(2) - 54698112000*x(0)^4*x(1)^2*x(2) -
1622817907200*x(1)^3*x(2) +     688552704000*x(0)*x(1)^3*x(2) -
302448384000*x(0)^2*x(1)^3*x(2) + 7150080000*x(0)^3*x(1)^3*x(2) -
411844608000*x(1)^4*x(2) + 34320384000*x(0)*x(1)^4*x(2) +     3434876247600*x(2)^2
+ 150348505104000*x(0)*x(2)^2 - 18526465994400*x(0)^2*x(2)^2 +
5540355676800*x(0)^3*x(2)^2 - 217183680000*x(0)^4*x(2)^2 -
10085441284800*x(1)*x(2)^2 + 1166228098560*x(0)*x(1)*x(2)^2 -
3221557920000*x(0)^2*x(1)*x(2)^2 - 1150269120000*x(0)^3*x(1)*x(2)^2 +
80438400000*x(0)^4*x(1)*x(2)^2 -     5685386112000*x(1)^2*x(2)^2 -
1737469440000*x(0)*x(1)^2*x(2)^2 + 965260800000*x(0)^2*x(1)^2*x(2)^2 +
223082496000*x(1)^3*x(2)^2 - 42900480000*x(0)*x(1)^3*x(2)^2 +
30528268848000*x(2)^3 - 2215273536000*x(0)*x(2)^3 + 260620416000*x(0)^2*x(2)^3 +
3764517120000*x(1)*x(2)^3 + 820471680000*x(0)*x(1)*x(2)^3 -
96526080000*x(0)^2*x(1)*x(2)^3 - 386104320000*x(1)^2*x(2)^3 - 450168248376000*x(3) +
362260413244800*x(0)*x(3) + 230868320256000*x(0)^2*x(3) -
191347785216000*x(0)^3*x(3) + 36889854624000*x(0)^4*x(3) - 1447891200000*x(0)^5*x(3) +
755972057030400*x(1)*x(3) - 1118635684992000*x(0)*x(1)*x(3) +
391382687808000*x(0)^2*x(1)*x(3) - 32947568640000*x(0)^3*x(1)*x(3) -
664957440000*x(0)^4*x(1)*x(3) + 53625600000*x(0)^5*x(1)*x(3) -
1768725504000*x(1)^2*x(3) -     13410690048000*x(0)*x(1)^2*x(3) -
11325726720000*x(0)^2*x(1)^2*x(3) + 1029611520000*x(0)^3*x(1)^2*x(3) -
16559585280000*x(1)^3*x(3) + 4232847360000*x(0)*x(1)^3*x(3) -
171601920000*x(0)^2*x(1)^3*x(3) - 196983529344000*x(2)*x(3) +
419412712320000*x(0)*x(2)*x(3) - 177772081536000*x(0)^2*x(2)*x(3) +
8108190720000*x(0)^3*x(2)*x(3) -     18960939648000*x(1)*x(2)*x(3) +
11776181760000*x(0)*x(1)*x(2)*x(3) + 7786437120000*x(0)^2*x(1)*x(2)*x(3) -
```

```
30030336000*x(0)^3*x(1)*x(2)*x(3) + 1647378432000*x(1)^2*x(2)*x(3) -
205922304000*x(0)*x(1)^2*x(2)*x(3) - 11440128000*x(1)^3*x(2)*x(3) +
1033794316800*x(2)^2*x(3) + 347493888000*x(0)*x(2)^2*x(3) -
398974464000*x(1)*x(2)^2*x(3) -     12870144000*x(0)*x(1)*x(2)^2*x(3) -
16047874483200*x(3)^2 + 14509800345600*x(0)*x(3)^2 - 926650368000*x(0)^2*x(3)^2 +
2496807936000*x(1)*x(3)^2 -     858009600000*x(0)*x(1)*x(3)^2 +
34320384000*x(0)^2*x(1)*x(3)^2 - 926650368000*x(2)*x(3)^2 +
34320384000*x(1)*x(2)*x(3)^2);(1/1488375)*(-31260501486963 + 19188051308244*x(0) +
41732561289078*x(0)^2 - 28455463927800*x(0)^3 - 2271003773220*x(0)^4 +
5919328270800*x(0)^5 -     1703105724600*x(0)^6 + 174410560800*x(0)^7 -
3619728000*x(0)^8 + 104257636328700*x(1) - 161054917560360*x(0)*x(1) +
1606903265520*x(0)^2*x(1) +     66688737937920*x(0)^3*x(1) -
30842436819600*x(0)^4*x(1) + 5239234526400*x(0)^5*x(1) - 278920152000*x(0)^6*x(1) -
3954888000*x(0)^7*x(1) +     134064000*x(0)^8*x(1) - 62986064732640*x(1)^2 +
159790227750720*x(0)*x(1)^2 - 115727564937600*x(0)^2*x(1)^2 +
33037452806400*x(0)^3*x(1)^2 -     3491267875200*x(0)^4*x(1)^2 +
17696448000*x(0)^5*x(1)^2 + 8043840000*x(0)^6*x(1)^2 - 10964215866240*x(1)^3 +
19834148275200*x(0)*x(1)^3 -     6206251564800*x(0)^2*x(1)^3 +
111541248000*x(0)^3*x(1)^3 + 58273152000*x(0)^4*x(1)^3 - 1072512000*x(0)^5*x(1)^3 +
2210385945600*x(1)^4 -     933514444800*x(0)*x(1)^4 + 308883456000*x(0)^2*x(1)^4 -
17160192000*x(0)^3*x(1)^4 + 274563072000*x(1)^5 - 45760512000*x(0)*x(1)^5 +
1906688000*x(0)^2*x(1)^5 -     26965270240956*x(2) + 65207817581760*x(0)*x(2) -
12895583505840*x(0)^2*x(2) - 24324003345600*x(0)^3*x(2) +
12543171958800*x(0)^4*x(2) -     1908320601600*x(0)^5*x(2) +
50676192000*x(0)^6*x(2) + 17687097610560*x(1)*x(2) - 102603589522560*x(0)*x(1)*x(2)
+ 92464641763200*x(0)^2*x(1)*x(2) -     26591406710400*x(0)^3*x(1)*x(2) +
2116334304000*x(0)^4*x(1)*x(2) + 40621392000*x(0)^5*x(1)*x(2) -
1876896000*x(0)^6*x(1)*x(2) + 14997601785600*x(1)^2*x(2) -
24906486528000*x(0)*x(1)^2*x(2) + 7958575296000*x(0)^2*x(1)^2*x(2) +
90091008000*x(0)^3*x(1)^2*x(2) - 54698112000*x(0)^4*x(1)^2*x(2) -
1622817907200*x(1)^3*x(2) +     688552704000*x(0)*x(1)^3*x(2) -
302448384000*x(0)^2*x(1)^3*x(2) + 7150080000*x(0)^3*x(1)^3*x(2) -
411844608000*x(1)^4*x(2) + 34320384000*x(0)*x(1)^4*x(2) +     3434876247600*x(2)^2
+ 150348505104000*x(0)*x(2)^2 - 18526465994400*x(0)^2*x(2)^2 +
5540355676800*x(0)^3*x(2)^2 - 2171836800000*x(0)^4*x(2)^2 -
10085441284800*x(1)*x(2)^2 + 11662280985600*x(0)*x(1)*x(2)^2 -
3221557920000*x(0)^2*x(1)*x(2)^2 - 1150269120000*x(0)^3*x(1)*x(2)^2 +
8043840000*x(0)^4*x(1)*x(2)^2 -     568538611200*x(1)^2*x(2)^2 -
1737469440000*x(0)*x(1)^2*x(2)^2 + 965260800000*x(0)^2*x(1)^2*x(2)^2 +
223082496000*x(1)^3*x(2)^2 - 42900480000*x(0)*x(1)^3*x(2)^2 +
30528268848000*x(2)^3 - 2215273536000*x(0)*x(2)^3 + 260620416000*x(0)^2*x(2)^3 +
3764517120000*x(1)*x(2)^3 + 820471680000*x(0)*x(1)*x(2)^3 -
96526080000*x(0)^2*x(1)*x(2)^3 - 386104320000*x(1)^2*x(2)^3 - 450168248376000*x(3) +
362260413244800*x(0)*x(3) + 230868320256000*x(0)^2*x(3) -
191347785216000*x(0)^3*x(3) + 36889854624000*x(0)^4*x(3) - 1447891200000*x(0)^5*x(3) +
755972057030400*x(1)*x(3) - 1118635684992000*x(0)*x(1)*x(3) +
391382687808000*x(0)^2*x(1)*x(3) - 32947568640000*x(0)^3*x(1)*x(3) -
664957440000*x(0)^4*x(1)*x(3) + 53625600000*x(0)^5*x(1)*x(3) -
1768725504000*x(1)^2*x(3) -     13410690048000*x(0)*x(1)^2*x(3) -
11325726720000*x(0)^2*x(1)^2*x(3) + 1029611520000*x(0)^3*x(1)^2*x(3) -
16559585280000*x(1)^3*x(3) + 4232847360000*x(0)*x(1)^3*x(3) -
171601920000*x(0)^2*x(1)^3*x(3) - 196983529344000*x(2)*x(3) +
419412712320000*x(0)*x(2)*x(3) - 177772081536000*x(0)^2*x(2)*x(3) +
8108190720000*x(0)^3*x(2)*x(3) -     18960939648000*x(1)*x(2)*x(3) +
11776181760000*x(0)*x(1)*x(2)*x(3) + 7786437120000*x(0)^2*x(1)*x(2)*x(3) -
30030336000*x(0)^3*x(1)*x(2)*x(3) + 1647378432000*x(1)^2*x(2)*x(3) -
205922304000*x(0)*x(1)^2*x(2)*x(3) - 11440128000*x(1)^3*x(2)*x(3) +
1033794316800*x(2)^2*x(3) + 347493888000*x(0)*x(2)^2*x(3) -
398974464000*x(1)*x(2)^2*x(3) -     12870144000*x(0)*x(1)*x(2)^2*x(3) -
16047874483200*x(3)^2 + 14509800345600*x(0)*x(3)^2 - 926650368000*x(0)^2*x(3)^2 +
2496807936000*x(1)*x(3)^2 -     858009600000*x(0)*x(1)*x(3)^2 +
34320384000*x(0)^2*x(1)*x(3)^2 - 926650368000*x(2)*x(3)^2 +
34320384000*x(1)*x(2)*x(3)^2);
> ideal i=f0,f1,f2,f3;
   ? error occurred in STDIN line 8: `ideal i=f0,f1,f2,f3;`
   ? wrong type declaration. type 'help ideal;'
   ? last reserved name was `ideal`
   skipping text from `=`
> ideal i=f0,f1,f2,f3;
```

```
> def T=solve(i,30,1,"nodisplay");

// 'solve' created a ring, in which a list SOL of numbers (the complex solutions)
// is stored.
// To access the list of complex solutions, type (if the name R was assigned
// to the return value):
       setring R; SOL;
> setring T;
> SOL[1][1];
[1]:
   [1]:
      (7.6986159167671201946604541112l2-i*0.994193183436608704326453171971)
   [2]:
      (1.990398350845398948768472013229-i*0.047787323516492258310104113l845)
   [3]:
      (15.47886822752413018045056230260l-i*5.164908972728780581045771308146)
   [4]:
      (35.588106358113085622278931533913-i*20.267200020113126002258171197882)
[2]:
   [1]:
      (7.6986159167671201946604541112l2+i*0.994193183436608704326453171971)
   [2]:
      (1.990398350845398948768472013229+i*0.047787323516492258310104113l845)
   [3]:
      (15.47886822752413018045056230260l+i*5.164908972728780581045771308146)
   [4]:
      (35.588106358113085622278931533913+i*20.267200020113126002258171197882)
[3]:
   [1]:
      (4.0653191751859081852499329292591-i*0.084651532185702109491269017726l)
   [2]:
      (3.3339374086668416647754844253620-i*0.114603163763464490185687107768)
   [3]:
      (3.5011443197110645961422522694910-i*0.132807994115193528796549504798)
   [4]:
      (3.2051632353500388125943261410530-i*0.149346269093064077609829376722)
[4]:
   [1]:
      (4.0653191751859081852499329292591+i*0.084651532185702109491269017726l)
   [2]:
      (3.3339374086668416647754844253620+i*0.114603163763464490185687107768)
   [3]:
      (3.5011443197110645961422522694910+i*0.132807994115193528796549504798)
   [4]:
      (3.2051632353500388125943261410530+i*0.149346269093064077609829376722)
[5]:
   [1]:
      (4.0439241181402846248344015869920-i*0.164600917315115697241872897572)
   [2]:
      (2.7988978367908390238814665148-i*0.015739205783219814262426835975l)
   [3]:
      (3.0008807038108828671261383510880-i*0.072801624327269662233923212546l)
   [4]:
      (2.5539642528810831793428168254110-i*0.029891843472506231993148561191)
[6]:
   [1]:
      (4.0439241181402846248344015869920+i*0.164600917315115697241872897572)
   [2]:
      (2.7988978367908390238814665148+i*0.015739205783219814262426835975l)
   [3]:
      (3.0008807038108828671261383510880+i*0.072801624327269662233923212546l)
   [4]:
      (2.5539642528810831793428168254110+i*0.029891843472506231993148561191)
[7]:
   [1]:
      (3.5463870871681824431121813412890+i*0.025497762957571226552573599321l)
   [2]:
      (2.7359346811840634643007304200540-i*0.020211373982733070853230490978)
   [3]:
```

```
      (2.780913720337302227255031099019-i*0.028262821018194709756590258620 4)
   [4]:
      (2.426248424263885443096373357794-i*0.029248961481446144511711379549 1)
[8]:
   [1]:
      (3.546387087168182443112181341289-i*0.025497762957571226552573599321 7)
   [2]:
      (2.735934681184063464300730420054+i*0.020211373982733070853230490978)
   [3]:
      (2.780913720337302227255031099019+i*0.028262821018194709756590258620 4)
   [4]:
      (2.426248424263885443096373357794+i*0.029248961481446144511711379549 1)
[9]:
   [1]:
      (3.074690766359377076310071570754-i*0.285142194651457744131264281589)
   [2]:
      (2.635988970097541882455615620044-i*0.009083202592245910699837442903 84)
   [3]:
      (2.468580389659578578122180207 9-i*0.038227949897923847179019615600 4)
   [4]:
      (2.242915961685472279391541163131-i*0.009415723608015531543490995471 89)
[10]:
   [1]:
      (3.074690766359377076310071570754+i*0.285142194651457744131264281589)
   [2]:
      (2.635988970097541882455615620044+i*0.009083202592245910699837442903 84)
   [3]:
      (2.468580389659578578122180207 9+i*0.038227949897923847179019615600 4)
   [4]:
      (2.242915961685472279391541163131+i*0.009415723608015531543490995471 89)
[11]:
   [1]:
      0.233120330250717764514117127227e+41
   [2]:
      -0.212672263142184072124657333684e+41
   [3]:
      0.981740358831002474672964451894e+40
   [4]:
      12248.235250569537731035592857834123
[12]:
   [1]:
      3.853266573064781836150861409443
   [2]:
      2.869867176071562056529149441141
   [3]:
      2.957130016015273172948942479216
   [4]:
      2.588450064418545975402315501117
[13]:
   [1]:
      3.930754648895702914249122440767
   [2]:
      1.340430792264410509828534664881
   [3]:
      2.957153015303767214447408316426
   [4]:
      2.542925564232708641889624529976
[14]:
   [1]:
      3.410399555922999865206180250 02
   [2]:
      2.639820227244579096147835645 8
   [3]:
      2.670787392912306521506906510255
   [4]:
      2.302508899108472548137588089622
[15]:
   [1]:
      3.731671937524934521428171714 31
```

```
   [2]:
      2.54976334886819161695775488734 5
   [3]:
      2.74995830287324766000044217054 7
   [4]:
      2.28852296032585958410828055722 9
[16]:
   [1]:
      2.92079652848823384574775234 84
   [2]:
      2.48552253474873746630398608886 2
   [3]:
      2.33226623515625304792362208
   [4]:
      2.05323669958137307059705672443 2
[17]:
   [1]:
      3.08498869819135998537040712948 3
   [2]:
      2.22932316366008051929964205339 3
   [3]:
      2.16932318264635628866561792943 7
   [4]:
      1.76159944686655568834571455441 4
[18]:
   [1]:
      3.36694433682619228958796074061 8
   [2]:
      2.06969906127709463622298530375
   [3]:
      2.26318284136230253791707489716 9
   [4]:
      1.71830277798523276749703751131 2
[19]:
   [1]:
      3.56038452832268423453118709712 6
   [2]:
      1.46210376752949515871811506659
   [3]:
      2.31514858126839832330221258046 9
   [4]:
      1.65156292030363159305670025945 6
[20]:
   [1]:
      2.78420244131005855482407721480 9
   [2]:
      2.17225258813700205801181213093 6
   [3]:
      1.98893605100488013916905192 67
   [4]:
      1.65154406028023772180165630622 4
[21]:
   [1]:
      2.94691985795260694233370280604 8
   [2]:
      2.05344409736002540491923185864 2
   [3]:
      1.96196944953647847751994990125
   [4]:
      1.53156373595662029826247155771 7
[22]:
   [1]:
      3.27322760511332632877575284298 4
   [2]:
      1.89448728725092719474854184883 9
   [3]:
      2.09625880948501693961051952227 8
   [4]:
      1.51611550263510135972486860 8
```

```
[23]:
   [1]:
      2.67074555189383274034966465356 6
   [2]:
      2.00229453940070839924906702192 2
   [3]:
      1.79658796579434101195290504424 7
   [4]:
      1.43168883272813740427123050624 4
[24]:
   [1]:
      2.46105254570787780262576773908 6
   [2]:
      1.95632955756985280408836951256 8
   [3]:
      1.69808792467181282781899578702 2
   [4]:
      1.36808679012908424247889473 45
[25]:
   [1]:
      2.32850219084381823991923756586 8
   [2]:
      1.78502735771013165373133832822 3
   [3]:
      1.49963376165062704740637443 72
   [4]:
      1.14559836979486810173568083367 5
[26]:
   [1]:
      2.02619544915365985375767039603 6
   [2]:
      1.48219293915888467537082459 42
   [3]:
      1.14281333712177840944703886 81
   [4]:
      0.75336320297273563904703795072 4
[27]:
   [1]:
      1.90441390190502292582387221637 5
   [2]:
      1.31216382708070065927241895611 4
   [3]:
      0.94631935764543300192415806693
   [4]:
      0.53255834822684671756620415359 6
>
```

16. Příloha E – výstup programu Mathematica

```
In[1]:= F1 = -858 * 3 + 792 * 3^3 + (506 * 3^5) / 3 - (88 * 3^7) / 9 - (176 * 3^9) / 315 +
          (32 * 3^11) / 945 + 792 * 3 * p2 - 572 * 3^3 * p2 - (88 * 3^5 * p2) / 3 + (528 * 3^7 * p2) / 35 -
          (704 * 3^9 * p2) / 945 - 286 * 3 * p2^2 + 88 * 3^3 * p2^2 - (528 * 3^5 * p2^2) / 5 +
          (704 * 3^7 * p2^2) / 315 + (88 * 3 * p2^3) / 3 + 176 * 3^3 * p2^3 + (704 * 3^5 * p2^3) / 45 -
          (352 * 3^3 * p2^4) / 9 + 2376 * p3 - (7832 * 3^2 * p3) / 3 - (6688 * 3^4 * p3) / 9 -
          (176 * 3^6 * p3) / 9 + (352 * 3^8 * p3) / 105 - 2376 * p2 * p3 + (6688 * 3^2 * p2 * p3) / 3 +
          (1936 * 3^4 * p2 * p3) / 3 - (704 * 3^6 * p2 * p3) / 45 + 1056 * p2^2 * p3 - 880 * 3^2 * p2^2 * p3 -
          (704 * 3^4 * p2^2 * p3) / 9 - (880 * p2^3 * p3) / 3 + (704 * 3^2 * p2^3 * p3) / 9 +
          (352 * p2^4 * p3) / 9 + (5984 * 3 * p3^2) / 9 - (2816 * 3^3 * p3^2) / 9 +
          (1408 * 3^5 * p3^2) / 45 - (1408 * 3 * p2 * p3^2) / 3 + (1408 * 3 * p2^2 * p3^2) / 9 +
          (2816 * p3^3) / 9 + (2816 * 3^2 * p3^3) / 27 - (2816 * p2 * p3^3) / 27 +
          220 * 3 * p4 + (1056 * 3^5 * p4) / 5 - (704 * 3^7 * p4) / 63 - 1056 * 3^3 * p2 * p4 +
          (704 * 3^3 * p2^2 * p4) / 3 - 1056 * p3 * p4 + 1056 * 3^2 * p3 * p4 - (704 * 3^4 * p3 * p4) / 9 +
          1056 * p2 * p3 * p4 - (704 * p2^2 * p3 * p4) / 3 - (2816 * 3 * p3^2 * p4) / 9 -
          (352 * 3 * p6) / 3 + 1408 * 3^3 * p6 - (2816 * 3^5 * p6) / 45 - (2816 * 3^3 * p2 * p6) / 9 -
          (2816 * p3 * p6) / 3 - (2816 * 3^2 * p3 * p6) / 9 + (2816 * p2 * p3 * p6) / 9;

In[2]:= F2 = 3471 * 3 - 4472 * 3^3 - 754 * 3^5 + (5512 * 3^7) / 45 + (52 * 3^9) / 35 -
          (416 * 3^11) / 945 + (32 * 3^13) / 4725 - 4680 * 3 * p2 + 4524 * 3^3 * p2 + 312 * 3^5 * p2 -
          (33904 * 3^7 * p2) / 315 + (9152 * 3^9 * p2) / 945 - (416 * 3^11 * p2) / 4725 +
          2314 * 3 * p2^2 - 1352 * 3^3 * p2^2 + (1768 * 3^5 * p2^2) / 3 - (9152 * 3^7 * p2^2) / 315 -
          (832 * 3^9 * p2^2) / 945 - (1352 * 3 * p2^3) / 3 - (2704 * 3^3 * p2^3) / 3 -
          (9152 * 3^5 * p2^3) / 45 + (832 * 3^7 * p2^3) / 105 + (52 * 3 * p2^4) / 3 +
          (4576 * 3^3 * p2^4) / 9 + (416 * 3^5 * p2^4) / 45 - (416 * 3^3 * p2^5) / 9 -
          9828 * p3 + 15704 * 3^2 * p3 + (32240 * 3^4 * p3) / 9 - (9568 * 3^6 * p3) / 45 -
          (2288 * 3^8 * p3) / 63 + (416 * 3^10 * p3) / 525 + 14040 * p2 * p3 - 17472 * 3^2 * p2 * p3 -
          (39104 * 3^4 * p2 * p3) / 9 + (832 * 3^6 * p2 * p3) / 3 - (416 * 3^8 * p2 * p3) / 315 -
          8112 * p2^2 * p3 + (25376 * 3^2 * p2^2 * p3) / 3 + (10400 * 3^4 * p2^2 * p3) / 9 -
          (832 * 3^6 * p2^2 * p3) / 45 + (8320 * p2^3 * p3) / 3 - (14144 * 3^2 * p2^3 * p3) / 9 -
          (832 * 3^4 * p2^3 * p3) / 9 - (5200 * p2^4 * p3) / 9 + (416 * 3^2 * p2^4 * p3) / 9 +
          (416 * p2^5 * p3) / 9 - (61984 * 3 * p3^2) / 9 + (15808 * 3^3 * p3^2) / 9 -
          (6656 * 3^5 * p3^2) / 45 + (6656 * 3^7 * p3^2) / 315 + (55744 * 3 * p2 * p3^2) / 9 -
          (3328 * 3^3 * p2 * p3^2) / 9 - (3328 * 3^5 * p2 * p3^2) / 45 - (23296 * 3 * p2^2 * p3^2) / 9 +
          (3328 * 3 * p2^3 * p3^2) / 9 - (11648 * p3^3) / 9 - (26624 * 3^2 * p3^3) / 27 +
          (3328 * 3^4 * p3^3) / 27 + (26624 * p2 * p3^3) / 27 + (6656 * 3^2 * p2 * p3^3) / 27 -
          (3328 * p2^2 * p3^3) / 27 - 52 * 3 * p4 - 416 * 3^3 * p4 - (3952 * 3^5 * p4) / 3 +
          (9152 * 3^7 * p4) / 63 - (1664 * 3^9 * p4) / 945 - 416 * 3 * p2 * p4 + 6240 * 3^3 * p2 * p4 -
          (1664 * 3^7 * p2 * p4) / 63 + 208 * 3 * p2^2 * p4 - (9152 * 3^3 * p2^2 * p4) / 3 +
          (1664 * 3^5 * p2^2 * p4) / 15 + (1664 * 3^3 * p2^3 * p4) / 9 + 2496 * p3 * p4 -
          (14144 * 3^2 * p3 * p4) / 3 + (11648 * 3^4 * p3 * p4) / 9 - (1664 * 3^6 * p3 * p4) / 45 -
          4992 * p2 * p3 * p4 - (1664 * 3^4 * p2 * p3 * p4) / 9 + (6656 * p2^2 * p3 * p4) / 3 +
          (1664 * 3^2 * p2^2 * p3 * p4) / 3 - (1664 * p2^3 * p3 * p4) / 9 + (26624 * 3 * p3^2 * p4) / 9 -
          (6656 * 3 * p2 * p3^2 * p4) / 9 - (6656 * p3^3 * p4) / 27 - 208 * 3 * p4^2 -
          (1664 * 3^5 * p4^2) / 15 + (1664 * 3^3 * p2 * p4^2) / 3 + 832 * p3 * p4^2 -
          (1664 * 3^2 * p3 * p4^2) / 3 - (1664 * p2 * p3 * p4^2) / 3 + (2912 * 3 * p6) / 3 -
          (24128 * 3^3 * p6) / 3 + (36608 * 3^5 * p6) / 45 - (3328 * 3^7 * p6) / 315 -
          (832 * 3 * p2 * p6) / 3 + (36608 * 3^3 * p2 * p6) / 9 - (6656 * 3^5 * p2 * p6) / 45 -
          (3328 * 3^3 * p2^2 * p6) / 9 + (11648 * p3 * p6) / 3 + (26624 * 3^2 * p3 * p6) / 9 -
          (3328 * 3^4 * p3 * p6) / 9 - (26624 * p2 * p3 * p6) / 9 - (6656 * 3^2 * p2 * p3 * p6) / 9 +
          (3328 * p2^2 * p3 * p6) / 9 - (6656 * 3^3 * p4 * p6) / 9 + (6656 * p3 * p4 * p6) / 9;

In[3]:= F3 = 27863 * 3 - 58820 * 3^3 - 2278 * 3^5 + (61744 * 3^7) / 15 - (16388 * 3^9) / 315 -
          (8704 * 3^11) / 675 + (3944 * 3^13) / 14175 - (1088 * 3^15) / 99225 + (64 * 3^17) / 99225 -
          67320 * 3 * p2 + 103428 * 3^3 * p2 + 3264 * 3^5 * p2 - (1211216 * 3^7 * p2) / 315 +
```

```
(387872 * 3^9 * p2) / 945 + (19856 * 3^11 * p2) / 4725 - (1088 * 3^15 * p2) / 33075 +
55726 * 3 * p2^2 - 61744 * 3^3 * p2^2 + (23800 * 3^5 * p2^2) / 3 - (85952 * 3^7 * p2^2) / 105 -
(44200 * 3^9 * p2^2) / 189 + (1088 * 3^11 * p2^2) / 315 + (1088 * 3^13 * p2^2) / 1575 -
21216 * 3 * p2^3 + (4624 * 3^3 * p2^3) / 3 - (31552 * 3^5 * p2^3) / 5 +
(173536 * 3^7 * p2^3) / 135 - (20672 * 3^11 * p2^3) / 2835 + (10268 * 3 * p2^4) / 3 +
(136000 * 3^3 * p2^4) / 9 + (11288 * 3^5 * p2^4) / 15 - (1088 * 3^7 * p2^4) / 7 +
(105536 * 3^9 * p2^4) / 2835 - (544 * 3 * p2^5) / 3 - (48688 * 3^3 * p2^5) / 9 -
(79424 * 3^7 * p2^5) / 945 + (136 * 3 * p2^6) / 9 + (5440 * 3^3 * p2^6) / 9 +
(11968 * 3^5 * p2^6) / 135 - (1088 * 3^3 * p2^7) / 27 - 80784 * p3 + 228072 * 3^2 * p3 +
(38488 * 3^4 * p3) / 3 - (54400 * 3^6 * p3) / 3 - (409088 * 3^8 * p3) / 315 +
(69632 * 3^10 * p3) / 2835 + (544 * 3^12 * p3) / 2835 + (14144 * 3^14 * p3) / 99225 +
201960 * p2 * p3 - 411536 * 3^2 * p2 * p3 - (464032 * 3^4 * p2 * p3) / 9 +
(106624 * 3^6 * p2 * p3) / 5 + (4352 * 3^8 * p2 * p3) / 21 - (18496 * 3^10 * p2 * p3) / 315 -
(1088 * 3^12 * p2 * p3) / 225 - 184008 * p2^2 * p3 + (885632 * 3^2 * p2^2 * p3) / 3 +
(103360 * 3^4 * p2^2 * p3) / 3 - (237184 * 3^6 * p2^2 * p3) / 45 +
(25568 * 3^8 * p2^2 * p3) / 45 + (96832 * 3^10 * p2^2 * p3) / 1575 +
(274720 * p2^3 * p3) / 3 - 104448 * 3^2 * p2^3 * p3 - (302464 * 3^4 * p2^3 * p3) / 27 -
(36992 * 3^6 * p2^3 * p3) / 27 - (64192 * 3^8 * p2^3 * p3) / 189 -
(266560 * p2^4 * p3) / 9 + 15232 * 3^2 * p2^4 * p3 + (28832 * 3^4 * p2^4 * p3) / 9 +
(105536 * 3^6 * p2^4 * p3) / 135 + (54400 * p2^5 * p3) / 9 - (1088 * 3^2 * p2^5 * p3) / 3 -
(18496 * 3^4 * p2^5 * p3) / 27 - (5984 * p2^6 * p3) / 9 - (1088 * 3^2 * p2^6 * p3) / 27 +
(1088 * p2^7 * p3) / 27 - 158848 * 3 * p3^2 + (170816 * 3^3 * p3^2) / 3 +
(56576 * 3^5 * p3^2) / 5 + (330752 * 3^7 * p3^2) / 135 + (126208 * 3^9 * p3^2) / 945 +
(8704 * 3^11 * p3^2) / 945 + (2227136 * 3 * p2 * p3^2) / 9 - (517888 * 3^3 * p2 * p3^2) / 9 -
(187136 * 3^5 * p2 * p3^2) / 9 - (765952 * 3^7 * p2 * p3^2) / 315 -
(17408 * 3^9 * p2 * p3^2) / 105 - 161024 * 3 * p2^2 * p3^2 + (226304 * 3^3 * p2^2 * p3^2) / 9 +
(461312 * 3^5 * p2^2 * p3^2) / 45 + (330752 * 3^7 * p2^2 * p3^2) / 315 +
(1492736 * 3 * p2^3 * p3^2) / 27 - (174080 * 3^3 * p2^3 * p3^2) / 27 -
(34816 * 3^5 * p2^3 * p3^2) / 15 - (82688 * 3 * p2^4 * p3^2) / 9 +
(8704 * 3^3 * p2^4 * p3^2) / 9 + (17408 * 3 * p2^5 * p3^2) / 27 - (95744 * p3^3) / 9 -
(761600 * 3^2 * p3^3) / 27 + (1118464 * 3^4 * p3^3) / 81 + (121856 * 3^6 * p3^3) / 81 +
(60928 * 3^8 * p3^3) / 405 + (526592 * p2 * p3^3) / 27 + (635392 * 3^2 * p2 * p3^3) / 27 -
(69632 * 3^4 * p2 * p3^3) / 9 - (34816 * 3^6 * p2 * p3^3) / 27 - (239360 * p2^2 * p3^3) / 27 -
(17408 * 3^2 * p2^2 * p3^3) / 3 + (17408 * 3^4 * p2^2 * p3^3) / 9 +
(34816 * p2^3 * p3^3) / 27 + (34816 * 3^2 * p2^3 * p3^3) / 81 - (8704 * p2^4 * p3^3) / 81 +
(17408 * 3 * p3^4) / 81 - (69632 * 3^3 * p3^4) / 81 + (278528 * 3^5 * p3^4) / 405 +
44132 * 3 * p4 - 47328 * 3^3 * p4 - 25840 * 3^5 * p4 + (1921408 * 3^7 * p4) / 315 -
(16048 * 3^9 * p4) / 945 - (2176 * 3^13 * p4) / 4725 - 49504 * 3 * p2 * p4 +
121312 * 3^3 * p2 * p4 - (1088 * 3^5 * p2 * p4) / 3 - (209984 * 3^7 * p2 * p4) / 45 +
(8704 * 3^9 * p2 * p4) / 63 + (23936 * 3^11 * p2 * p4) / 1575 + 24752 * 3 * p2^2 * p4 -
(324224 * 3^3 * p2^2 * p4) / 3 + (227936 * 3^5 * p2^2 * p4) / 15 -
(56576 * 3^9 * p2^2 * p4) / 315 - (11968 * 3 * p2^3 * p4) / 3 + (201280 * 3^3 * p2^3 * p4) / 9 -
(8704 * 3^5 * p2^3 * p4) / 3 + (726784 * 3^7 * p2^3 * p4) / 945 - (272 * 3 * p2^4 * p4) / 3 -
(15232 * 3^5 * p2^4 * p4) / 15 + (2176 * 3^3 * p2^5 * p4) / 9 - 98736 * p3 * p4 +
77248 * 3^2 * p3 * p4 + (634304 * 3^4 * p3 * p4) / 9 - (298112 * 3^6 * p3 * p4) / 45 -
(242624 * 3^8 * p3 * p4) / 315 - (28288 * 3^10 * p3 * p4) / 525 + 58752 * p2 * p3 * p4 -
(315520 * 3^2 * p2 * p3 * p4) / 3 - (263296 * 3^4 * p2 * p3 * p4) / 9 +
(134912 * 3^6 * p2 * p3 * p4) / 15 + (263296 * 3^8 * p2 * p3 * p4) / 315 +
(14144 * p2^2 * p3 * p4) / 3 + (232832 * 3^2 * p2^2 * p3 * p4) / 3 -
(23936 * 3^4 * p2^2 * p3 * p4) / 3 - (187136 * 3^6 * p2^2 * p3 * p4) / 45 -
(41344 * p2^3 * p3 * p4) / 9 - (47872 * 3^2 * p2^3 * p3 * p4) / 3 +
(134912 * 3^4 * p2^3 * p3 * p4) / 27 + (1088 * p2^4 * p3 * p4) / 3 +
(2176 * 3^2 * p2^4 * p3 * p4) / 3 - (2176 * p2^5 * p3 * p4) / 9 + (239360 * 3 * p3^2 * p4) / 9 -
(95744 * 3^3 * p3^2 * p4) / 9 - (226304 * 3^5 * p3^2 * p4) / 45 -
(34816 * 3^7 * p3^2 * p4) / 45 - (295936 * 3 * p2 * p3^2 * p4) / 9 +
```

```
(34816 * 3^3 * p2 * p3^2 * p4) / 3 + (243712 * 3^5 * p2 * p3^2 * p4) / 45 +
(17408 * 3 * p2^2 * p3^2 * p4) / 3 - (34816 * 3^3 * p2^2 * p3^2 * p4) / 9 -
(34816 * 3 * p2^3 * p3^2 * p4) / 27 - (478720 * p3^3 * p4) / 27 - (34816 * 3^2 * p3^3 * p4) / 9 -
(69632 * 3^4 * p3^3 * p4) / 27 + (69632 * p2 * p3^3 * p4) / 9 - 8432 * 3 * p4^2 -
2176 * 3^3 * p4^2 - (66912 * 3^5 * p4^2) / 5 + (21760 * 3^7 * p4^2) / 21 +
(82688 * 3^9 * p4^2) / 945 - 2176 * 3 * p2 * p4^2 + (194752 * 3^3 * p2 * p4^2) / 3 -
(126208 * 3^7 * p2 * p4^2) / 105 + 1632 * 3 * p2^2 * p4^2 - 21760 * 3^3 * p2^2 * p4^2 +
(56576 * 3^5 * p2^2 * p4^2) / 15 - (4352 * 3^3 * p2^3 * p4^2) / 9 + 50048 * p3 * p4^2 -
56576 * 3^2 * p3 * p4^2 + (23936 * 3^4 * p3 * p4^2) / 3 + (47872 * 3^6 * p3 * p4^2) / 45 -
(178432 * p2 * p3 * p4^2) / 3 + 4352 * 3^2 * p2 * p3 * p4^2 - (21760 * 3^4 * p2 * p3 * p4^2) / 3 +
15232 * p2^2 * p3 * p4^2 + (4352 * 3^2 * p2^2 * p3 * p4^2) / 3 + (4352 * p2^3 * p3 * p4^2) / 9 +
17408 * 3 * p3^2 * p4^2 + (34816 * 3^3 * p3^2 * p4^2) / 9 - (34816 * p3^3 * p4^2) / 27 -
1088 * 3 * p4^3 - (8704 * 3^5 * p4^3) / 15 + (8704 * 3^3 * p2 * p4^3) / 3 +
4352 * p3 * p4^3 - (8704 * 3^2 * p3 * p4^3) / 3 - (8704 * p2 * p3 * p4^3) / 3 - 14144 * 3 * p6 -
(260032 * 3^3 * p6) / 3 + (530944 * 3^5 * p6) / 15 + (4352 * 3^7 * p6) / 135 -
(8704 * 3^11 * p6) / 2835 + (31552 * 3 * p2 * p6) / 3 + (1048832 * 3^3 * p2 * p6) / 9 -
(322048 * 3^5 * p2 * p6) / 15 + (17408 * 3^7 * p2 * p6) / 21 + (34816 * 3^9 * p2 * p6) / 405 -
(17408 * 3 * p2^2 * p6) / 3 - (369920 * 3^3 * p2^2 * p6) / 9 - (713728 * 3^7 * p2^2 * p6) / 945 +
(8704 * 3 * p2^3 * p6) / 9 + (87040 * 3^3 * p2^3 * p6) / 9 + (243712 * 3^5 * p2^3 * p6) / 135 -
(43520 * 3^3 * p2^4 * p6) / 27 + (330752 * p3 * p6) / 3 - (4352 * 3^2 * p3 * p6) / 3 -
(1779968 * 3^4 * p3 * p6) / 27 - (139264 * 3^6 * p3 * p6) / 27 - (322048 * 3^8 * p3 * p6) / 945 -
(1231616 * p2 * p3 * p6) / 9 + (8704 * 3^2 * p2 * p3 * p6) / 3 + (400384 * 3^4 * p2 * p3 * p6) / 9 +
(452608 * 3^6 * p2 * p3 * p6) / 135 + (552704 * p2^2 * p3 * p6) / 9 -
(34816 * 3^2 * p2^2 * p3 * p6) / 3 - (226304 * 3^4 * p2^2 * p3 * p6) / 27 -
(121856 * p2^3 * p3 * p6) / 9 + (34816 * 3^2 * p2^3 * p3 * p6) / 27 +
(43520 * p2^4 * p3 * p6) / 27 + (557056 * 3 * p3^2 * p6) / 27 -
(139264 * 3^3 * p3^2 * p6) / 27 - (139264 * 3^5 * p3^2 * p6) / 45 -
(139264 * 3 * p2 * p3^2 * p6) / 9 + (139264 * 3 * p2^2 * p3^2 * p6) / 27 +
(278528 * p3^3 * p6) / 27 + (278528 * 3^2 * p3^3 * p6) / 81 -
(278528 * p2 * p3^3 * p6) / 81 + (47872 * 3 * p4 * p6) / 3 - (765952 * 3^3 * p4 * p6) / 9 +
(34816 * 3^5 * p4 * p6) / 3 + (1114112 * 3^7 * p4 * p6) / 945 - (8704 * 3 * p2 * p4 * p6) / 3 -
(139264 * 3^5 * p2 * p4 * p6) / 15 + (69632 * 3^3 * p2^2 * p4 * p6) / 9 +
(165376 * p3 * p4 * p6) / 9 + (139264 * 3^2 * p3 * p4 * p6) / 3 +
(139264 * 3^4 * p3 * p4 * p6) / 27 + (34816 * p2 * p3 * p4 * p6) / 3 -
(69632 * p2^2 * p3 * p4 * p6) / 9 - (278528 * 3 * p3^2 * p4 * p6) / 27 -
(34816 * 3^3 * p4^2 * p6) / 9 + (34816 * p3 * p4^2 * p6) / 9 - (17408 * 3 * p6^2) / 9 +
(348160 * 3^3 * p6^2) / 9 + (557056 * 3^5 * p6^2) / 135 - (278528 * 3^3 * p2 * p6^2) / 27 -
(278528 * p3 * p6^2) / 9 - (278528 * 3^2 * p3 * p6^2) / 27 + (278528 * p2 * p3 * p6^2) / 27;
```

In[4]:=

```
F4 = -63973 * 3 + 204136 * 3^3 - 31958 * 3^5 - 26144 * 3^7 + (7220 * 3^9) / 21 +
(1133312 * 3^11) / 4725 + (19304 * 3^13) / 14175 - (608 * 3^15) / 675 -
(1216 * 3^17) / 99225 + (2176 * 3^19) / 1488375 + 195624 * 3 * p2 -
416252 * 3^3 * p2 + (108680 * 3^5 * p2) / 3 + (8034416 * 3^7 * p2) / 315 -
(412832 * 3^9 * p2) / 135 - (587632 * 3^11 * p2) / 4725 + (372704 * 3^13 * p2) / 14175 +
(20672 * 3^15 * p2) / 33075 - (111872 * 3^17 * p2) / 1488375 - 201058 * 3 * p2^2 +
305216 * 3^3 * p2^2 - (659528 * 3^5 * p2^2) / 15 + (32832 * 3^7 * p2^2) / 5 +
(2519096 * 3^9 * p2^2) / 945 - (134368 * 3^11 * p2^2) / 675 - (20672 * 3^13 * p2^2) / 1575 +
(2120704 * 3^15 * p2^2) / 1488375 + 96824 * 3 * p2^3 - (165680 * 3^3 * p2^3) / 3 +
(263872 * 3^5 * p2^3) / 15 - (1777184 * 3^7 * p2^3) / 135 + (11552 * 3^9 * p2^3) / 189 +
(392768 * 3^11 * p2^3) / 2835 - (4864 * 3^13 * p2^3) / 405 - (71060 * 3 * p2^4) / 3 -
(475456 * 3^3 * p2^4) / 9 + (37544 * 3^5 * p2^4) / 9 + (2492192 * 3^7 * p2^4) / 945 -
(2005184 * 3^9 * p2^4) / 2835 + (63232 * 3^11 * p2^4) / 1575 + (10336 * 3 * p2^5) / 3 +
(272080 * 3^3 * p2^5) / 9 - 1824 * 3^5 * p2^5 + (1509056 * 3^7 * p2^5) / 945 +
(4864 * 3^9 * p2^5) / 945 - (5624 * 3 * p2^6) / 9 - 5472 * 3^3 * p2^6 -
(227392 * 3^5 * p2^6) / 135 - (252928 * 3^7 * p2^6) / 945 + (608 * 3 * p2^7) / 9 +
```

```
(20672 * 3^3 * p2^7) / 27 + (53504 * 3^5 * p2^7) / 135 - (2432 * 3^3 * p2^8) / 27 +
186732 * p3 - 884488 * 3^2 * p3 + (535952 * 3^4 * p3) / 3 + (6914176 * 3^6 * p3) / 45 +
(3031792 * 3^8 * p3) / 315 - (16863488 * 3^10 * p3) / 14175 - (179968 * 3^12 * p3) / 1575 +
(1216 * 3^14 * p3) / 945 + (89984 * 3^16 * p3) / 297675 - 586872 * p2 * p3 +
(5413024 * 3^2 * p2 * p3) / 3 - (1202624 * 3^4 * p2 * p3) / 9 - (9085952 * 3^6 * p2 * p3) / 45 -
(114304 * 3^8 * p2 * p3) / 63 + (24366208 * 3^10 * p2 * p3) / 14175 -
(37696 * 3^12 * p2 * p3) / 2835 - (3010816 * 3^14 * p2 * p3) / 297675 + 652080 * p2^2 * p3 -
(4474880 * 3^2 * p2^2 * p3) / 3 + 38304 * 3^4 * p2^2 * p3 + (2701952 * 3^6 * p2^2 * p3) / 45 -
(194560 * 3^8 * p2^2 * p3) / 21 - (13376 * 3^10 * p2^2 * p3) / 27 +
(651776 * 3^12 * p2^2 * p3) / 6075 - (1153984 * p2^3 * p3) / 3 + 622592 * 3^2 * p2^3 * p3 +
(1111424 * 3^4 * p2^3 * p3) / 27 + (3263744 * 3^6 * p2^3 * p3) / 135 +
(1173440 * 3^8 * p2^3 * p3) / 189 - (5306624 * 3^10 * p2^3 * p3) / 14175 +
(1351888 * p2^4 * p3) / 9 - (1096832 * 3^2 * p2^4 * p3) / 9 - (1308416 * 3^4 * p2^4 * p3) / 27 -
(2678848 * 3^6 * p2^4 * p3) / 135 - (199424 * 3^8 * p2^4 * p3) / 945 -
(369664 * p2^5 * p3) / 9 + (17024 * 3^2 * p2^5 * p3) / 3 + (548416 * 3^4 * p2^5 * p3) / 27 +
(345344 * 3^6 * p2^5 * p3) / 135 + (68096 * p2^6 * p3) / 9 + (27968 * 3^2 * p2^6 * p3) / 27 -
(68096 * 3^4 * p2^6 * p3) / 27 - (27968 * p2^7 * p3) / 27 - (4864 * 3^2 * p2^7 * p3) / 27 +
(2432 * p2^8 * p3) / 27 + (2399776 * 3 * p3^2) / 3 - (4218304 * 3^3 * p3^2) / 9 -
(2434432 * 3^5 * p3^2) / 15 - (3346432 * 3^7 * p3^2) / 135 - (469376 * 3^9 * p3^2) / 945 +
(19456 * 3^11 * p3^2) / 81 + (768512 * 3^13 * p3^2) / 42525 - (13386944 * 3 * p2 * p3^2) / 9 +
(2033152 * 3^3 * p2 * p3^2) / 3 + (4703488 * 3^5 * p2 * p3^2) / 15 +
(34271744 * 3^7 * p2 * p3^2) / 945 - (1177088 * 3^9 * p2 * p3^2) / 2835 -
(12918784 * 3^11 * p2 * p3^2) / 42525 + 1189248 * 3 * p2^2 * p3^2 -
466944 * 3^3 * p2^2 * p3^2 - (9343744 * 3^5 * p2^2 * p3^2) / 45 -
(6148096 * 3^7 * p2^2 * p3^2) / 315 + (301568 * 3^9 * p2^2 * p3^2) / 243 -
(14421760 * 3 * p2^3 * p3^2) / 27 + (4893184 * 3^3 * p2^3 * p3^2) / 27 +
(9202688 * 3^5 * p2^3 * p3^2) / 135 + (1011712 * 3^7 * p2^3 * p3^2) / 945 +
(3806080 * 3 * p2^4 * p3^2) / 27 - (1070080 * 3^3 * p2^4 * p3^2) / 27 -
(943616 * 3^5 * p2^4 * p3^2) / 135 - (593408 * 3 * p2^5 * p3^2) / 27 +
(77824 * 3^3 * p2^5 * p3^2) / 27 + (48640 * 3 * p2^6 * p3^2) / 27 + (1009280 * p3^3) / 9 +
(2937856 * 3^2 * p3^3) / 27 - (13624064 * 3^4 * p3^3) / 81 - (2957312 * 3^6 * p3^3) / 135 +
(2607104 * 3^8 * p3^3) / 2835 + (10875904 * 3^10 * p3^3) / 42525 -
(4795904 * p2 * p3^3) / 27 - (3453440 * 3^2 * p2 * p3^3) / 27 +
(11517952 * 3^4 * p2 * p3^3) / 81 + (311296 * 3^6 * p2 * p3^3) / 15 -
(15545344 * 3^8 * p2 * p3^3) / 8505 + (2913536 * p2^2 * p3^3) / 27 +
(1595392 * 3^2 * p2^2 * p3^3) / 27 - (428032 * 3^4 * p2^2 * p3^3) / 9 -
(1595392 * 3^6 * p2^2 * p3^3) / 1215 - (894976 * p2^3 * p3^3) / 27 -
(1478656 * 3^2 * p2^3 * p3^3) / 81 + (272384 * 3^4 * p2^3 * p3^3) / 81 +
(544768 * p2^4 * p3^3) / 81 + (97280 * 3^2 * p2^4 * p3^3) / 27 -
(19456 * p2^5 * p3^3) / 27 + (3716096 * 3 * p3^4) / 81 + (155648 * 3^3 * p3^4) / 9 -
(311296 * 3^5 * p3^4) / 45 + (9961472 * 3^7 * p3^4) / 8505 - (4046848 * 3 * p2 * p3^4) / 81 -
(1245184 * 3^3 * p2 * p3^4) / 81 - (311296 * 3^5 * p2 * p3^4) / 1215 +
(622592 * 3 * p2^2 * p3^4) / 27 + (1245184 * 3^3 * p2^2 * p3^4) / 243 -
(311296 * 3 * p2^3 * p3^4) / 81 + (311296 * p3^5) / 27 + (622592 * 3^2 * p3^5) / 81 +
(311296 * 3^4 * p3^5) / 243 - (622592 * p2 * p3^5) / 81 - (622592 * 3^2 * p2 * p3^5) / 243 +
(311296 * p2^2 * p3^5) / 243 - 206492 * 3 * p4 + 244720 * 3^3 * p4 + (1419376 * 3^5 * p4) / 15 -
(1847104 * 3^7 * p4) / 45 - (953648 * 3^9 * p4) / 945 + (1704832 * 3^11 * p4) / 4725 +
(41344 * 3^13 * p4) / 4725 - (530176 * 3^15 * p4) / 496125 + 276640 * 3 * p2 * p4 -
532000 * 3^3 * p2 * p4 + (187264 * 3^5 * p2 * p4) / 3 + (2019776 * 3^7 * p2 * p4) / 45 -
(294272 * 3^9 * p2 * p4) / 105 - (454784 * 3^11 * p2 * p4) / 1575 +
(447488 * 3^13 * p2 * p4) / 14175 - 138320 * 3 * p2^2 * p4 + (1457984 * 3^3 * p2^2 * p4) / 3 -
(415264 * 3^5 * p2^2 * p4) / 3 - (165376 * 3^7 * p2^2 * p4) / 45 +
(1074944 * 3^9 * p2^2 * p4) / 315 - (1250048 * 3^11 * p2^2 * p4) / 4725 +
(63232 * 3 * p2^3 * p4) / 3 - (1212352 * 3^3 * p2^3 * p4) / 9 + (340480 * 3^5 * p2^3 * p4) / 9 -
(13808896 * 3^7 * p2^3 * p4) / 945 + (1439744 * 3^9 * p2^3 * p4) / 2835 +
```

```
4560 * 3 * p2^4 * p4 + (2432 * 3^3 * p2^4 * p4) / 3 + (289408 * 3^5 * p2^4 * p4) / 15 +
(228608 * 3^7 * p2^4 * p4) / 135 - (2432 * 3 * p2^5 * p4) / 3 - (41344 * 3^3 * p2^5 * p4) / 9 -
(214016 * 3^5 * p2^5 * p4) / 45 + (34048 * 3^3 * p2^6 * p4) / 27 +
521664 * p3 * p4 - (1578368 * 3^2 * p3 * p4) / 3 - (3584768 * 3^4 * p3 * p4) / 9 +
(3339136 * 3^6 * p3 * p4) / 45 + (552064 * 3^8 * p3 * p4) / 45 - (396416 * 3^10 * p3 * p4) / 945 -
(1639168 * 3^12 * p3 * p4) / 14175 - 569088 * p2 * p3 * p4 + (2380928 * 3^2 * p2 * p3 * p4) / 3 +
(435328 * 3^4 * p2 * p3 * p4) / 9 - (6148096 * 3^6 * p2 * p3 * p4) / 45 -
(201856 * 3^8 * p2 * p3 * p4) / 21 + (19981312 * 3^10 * p2 * p3 * p4) / 14175 +
(442624 * p2^2 * p3 * p4) / 3 - (1651328 * 3^2 * p2^2 * p3 * p4) / 3 +
(2018560 * 3^4 * p2^2 * p3 * p4) / 9 + (1405696 * 3^6 * p2^2 * p3 * p4) / 15 -
(938752 * 3^8 * p2^2 * p3 * p4) / 315 + (99712 * p2^3 * p3 * p4) / 9 +
(1439744 * 3^2 * p2^3 * p3 * p4) / 9 - (3847424 * 3^4 * p2^3 * p3 * p4) / 27 -
(1439744 * 3^6 * p2^3 * p3 * p4) / 135 - (46208 * p2^4 * p3 * p4) / 3 -
17024 * 3^2 * p2^4 * p3 * p4 + (530176 * 3^4 * p2^4 * p3 * p4) / 27 +
(70528 * p2^5 * p3 * p4) / 9 + (19456 * 3^2 * p2^5 * p3 * p4) / 9 -
(34048 * p2^6 * p3 * p4) / 27 - (1454336 * 3 * p3^2 * p4) / 9 + (807424 * 3^3 * p3^2 * p4) / 3 +
(3336704 * 3^5 * p3^2 * p4) / 45 + (2023424 * 3^7 * p3^2 * p4) / 315 -
(1381376 * 3^9 * p3^2 * p4) / 945 + (3054592 * 3 * p2 * p3^2 * p4) / 9 -
(2315264 * 3^3 * p2 * p3^2 * p4) / 9 - (311296 * 3^5 * p2 * p3^2 * p4) / 3 +
(1245184 * 3^7 * p2 * p3^2 * p4) / 189 - (1819136 * 3 * p2^2 * p3^2 * p4) / 9 +
(311296 * 3^3 * p2^2 * p3^2 * p4) / 3 + (38912 * 3^5 * p2^2 * p3^2 * p4) / 3 +
(1945600 * 3 * p2^3 * p3^2 * p4) / 27 - (155648 * 3^3 * p2^3 * p3^2 * p4) / 27 -
(330752 * 3 * p2^4 * p3^2 * p4) / 27 + (1099264 * p3^3 * p4) / 27 +
(38912 * 3^2 * p3^3 * p4) / 27 + (972800 * 3^4 * p3^3 * p4) / 27 -
(2256896 * 3^6 * p3^3 * p4) / 405 - (38912 * p2 * p3^3 * p4) / 9 +
(389120 * 3^2 * p2 * p3^3 * p4) / 9 + (77824 * 3^4 * p2 * p3^3 * p4) / 81 -
(194560 * p2^2 * p3^3 * p4) / 9 - (544768 * 3^2 * p2^2 * p3^3 * p4) / 27 +
(389120 * p2^3 * p3^3 * p4) / 81 - (622592 * 3 * p3^4 * p4) / 27 -
(622592 * 3^3 * p3^4 * p4) / 81 + (622592 * 3 * p2 * p3^4 * p4) / 81 - 7600 * 3 * p4^2 +
41344 * 3^3 * p4^2 + (234080 * 3^5 * p4^2) / 3 - (566656 * 3^7 * p4^2) / 63 -
(1571072 * 3^9 * p4^2) / 945 + (184832 * 3^11 * p4^2) / 945 + 41344 * 3 * p2 * p4^2 -
(1088320 * 3^3 * p2 * p4^2) / 3 - 2432 * 3^5 * p2 * p4^2 + (2397952 * 3^7 * p2 * p4^2) / 105 -
(1420288 * 3^9 * p2 * p4^2) / 945 - 28576 * 3 * p2^2 * p4^2 + 192128 * 3^3 * p2^2 * p4^2 -
(1074944 * 3^5 * p2^2 * p4^2) / 15 - (19456 * 3^7 * p2^2 * p4^2) / 315 +
2432 * 3 * p2^3 * p4^2 + (82688 * 3^3 * p2^3 * p4^2) / 9 + (214016 * 3^5 * p2^3 * p4^2) / 15 -
(48640 * 3^3 * p2^4 * p4^2) / 9 - 115520 * p3 * p4^2 + (588544 * 3^2 * p3 * p4^2) / 3 -
(719872 * 3^4 * p3 * p4^2) / 9 - (958208 * 3^6 * p3 * p4^2) / 45 +
(651776 * 3^8 * p3 * p4^2) / 315 + (734464 * p2 * p3 * p4^2) / 3 -
29184 * 3^2 * p2 * p3 * p4^2 + (442624 * 3^4 * p2 * p3 * p4^2) / 3 -
(252928 * 3^6 * p2 * p3 * p4^2) / 45 - 77824 * p2^2 * p3 * p4^2 -
(53504 * 3^2 * p2^2 * p3 * p4^2) / 3 - (214016 * 3^4 * p2^2 * p3 * p4^2) / 9 -
(170240 * p2^3 * p3 * p4^2) / 9 - (19456 * 3^2 * p2^3 * p3 * p4^2) / 3 +
(48640 * p2^4 * p3 * p4^2) / 9 - (515584 * 3 * p3^2 * p4^2) / 9 -
77824 * 3^3 * p3^2 * p4^2 + (38912 * 3^5 * p3^2 * p4^2) / 5 - 38912 * 3 * p2 * p3^2 * p4^2 +
(194560 * 3 * p2^2 * p3^2 * p4^2) / 9 + (544768 * p3^3 * p4^2) / 27 +
(389120 * 3^2 * p3^3 * p4^2) / 27 - (77824 * p2 * p3^3 * p4^2) / 27 + 15808 * 3 * p4^3 +
(165376 * 3^5 * p4^3) / 15 - (19456 * 3^7 * p4^3) / 63 - (165376 * 3^3 * p2 * p4^3) / 3 +
(19456 * 3^3 * p2^2 * p4^3) / 3 - 68096 * p3 * p4^3 + (165376 * 3^2 * p3 * p4^3) / 3 -
(19456 * 3^4 * p3 * p4^3) / 9 + (165376 * p2 * p3 * p4^3) / 3 - (19456 * p2^2 * p3 * p4^3) / 3 -
(77824 * 3 * p3^2 * p4^3) / 9 + 165984 * 3 * p6 + (509504 * 3^3 * p6) / 3 -
(3827968 * 3^5 * p6) / 15 - (1113856 * 3^7 * p6) / 135 + (435328 * 3^9 * p6) / 189 +
(165376 * 3^11 * p6) / 2835 - (19456 * 3^13 * p6) / 2835 - (614080 * 3 * p2 * p6) / 3 -
(3147008 * 3^3 * p2 * p6) / 9 + (1858048 * 3^5 * p2 * p6) / 9 - (5846528 * 3^7 * p2 * p6) / 945 -
(661504 * 3^9 * p2 * p6) / 405 + (797696 * 3^11 * p2 * p6) / 4725 +
(330752 * 3 * p2^2 * p6) / 3 + (1716992 * 3^3 * p2^2 * p6) / 9 -
```

```
(53504 * 3^5 * p2^2 * p6) / 3 + (13560832 * 3^7 * p2^2 * p6) / 945 -
(38912 * 3^9 * p2^2 * p6) / 45 - (243200 * 3 * p2^3 * p6) / 9 - (243200 * 3^3 * p2^3 * p6) / 3 -
(4630528 * 3^5 * p2^3 * p6) / 135 - (194560 * 3^7 * p2^3 * p6) / 189 +
(17024 * 3 * p2^4 * p6) / 9 + (826880 * 3^3 * p2^4 * p6) / 27 +
(914432 * 3^5 * p2^4 * p6) / 135 - (97280 * 3^3 * p2^5 * p6) / 27 - (1928576 * p3 * p6) / 3 +
301568 * 3^2 * p3 * p6 + (18507520 * 3^4 * p3 * p6) / 27 + (9766912 * 3^6 * p3 * p6) / 135 -
(369664 * 3^8 * p3 * p6) / 135 - (369664 * 3^10 * p3 * p6) / 525 + (8735744 * p2 * p3 * p6) / 9 -
(3317248 * 3^2 * p2 * p3 * p6) / 9 - (17821696 * 3^4 * p2 * p3 * p6) / 27 -
(9261056 * 3^6 * p2 * p3 * p6) / 135 + (4416512 * 3^8 * p2 * p3 * p6) / 945 -
(5189888 * p2^2 * p3 * p6) / 9 + 233472 * 3^2 * p2^2 * p3 * p6 +
(6225920 * 3^4 * p2^2 * p3 * p6) / 27 + (1128448 * 3^6 * p2^2 * p3 * p6) / 135 +
(1673216 * p2^3 * p3 * p6) / 9 - (1245184 * 3^2 * p2^3 * p3 * p6) / 27 -
(661504 * 3^4 * p2^3 * p3 * p6) / 27 - (1031168 * p2^4 * p3 * p6) / 27 -
(19456 * 3^2 * p2^4 * p3 * p6) / 27 + (97280 * p2^5 * p3 * p6) / 27 -
(9494528 * 3 * p3^2 * p6) / 27 + (856064 * 3^3 * p3^2 * p6) / 27 +
(6848512 * 3^5 * p3^2 * p6) / 135 - (5292032 * 3^7 * p3^2 * p6) / 945 +
(9261056 * 3 * p2 * p3^2 * p6) / 27 + (311296 * 3^3 * p2 * p3^2 * p6) / 27 -
(1245184 * 3^5 * p2 * p3^2 * p6) / 135 - (4046848 * 3 * p2^2 * p3^2 * p6) / 27 -
(311296 * 3^3 * p2^2 * p3^2 * p6) / 27 + (622592 * 3 * p2^3 * p3^2 * p6) / 27 -
(2957312 * p3^3 * p6) / 27 - (6225920 * 3^2 * p3^3 * p6) / 81 -
(311296 * 3^4 * p3^3 * p6) / 81 + (6225920 * p2 * p3^3 * p6) / 81 +
(622592 * 3^2 * p2 * p3^3 * p6) / 27 - (311296 * p2^2 * p3^3 * p6) / 27 -
(252928 * 3 * p4 * p6) / 3 + (3930112 * 3^3 * p4 * p6) / 9 - (924160 * 3^5 * p4 * p6) / 9 -
(21168128 * 3^7 * p4 * p6) / 945 + (1011712 * 3^9 * p4 * p6) / 405 + 29184 * 3 * p2 * p4 * p6 +
(19456 * 3^3 * p2 * p4 * p6) / 3 + (2646016 * 3^5 * p2 * p4 * p6) / 15 -
(6459392 * 3^7 * p2 * p4 * p6) / 945 + (9728 * 3 * p2^2 * p4 * p6) / 3 -
(1323008 * 3^3 * p2^2 * p4 * p6) / 9 - (1478656 * 3^5 * p2^2 * p4 * p6) / 45 +
(544768 * 3^3 * p2^3 * p4 * p6) / 27 - (398848 * p3 * p4 * p6) / 9 -
(2607104 * 3^2 * p3 * p4 * p6) / 9 - (2762752 * 3^4 * p3 * p4 * p6) / 27 +
(1789952 * 3^6 * p3 * p4 * p6) / 135 - (428032 * p2 * p3 * p4 * p6) / 3 -
(77824 * 3^2 * p2 * p3 * p4 * p6) / 3 + (389120 * 3^4 * p2 * p3 * p4 * p6) / 27 +
(1206272 * p2^2 * p3 * p4 * p6) / 9 + (389120 * 3^2 * p2^2 * p3 * p4 * p6) / 9 -
(544768 * p2^3 * p3 * p4 * p6) / 27 + (4358144 * 3 * p3^2 * p4 * p6) / 27 +
(622592 * 3^3 * p3^2 * p4 * p6) / 27 - (1245184 * 3 * p2 * p3^2 * p4 * p6) / 27 -
(622592 * p3^3 * p4 * p6) / 81 - 9728 * 3 * p4^2 * p6 + (661504 * 3^3 * p4^2 * p6) / 9 -
(77824 * 3^5 * p4^2 * p6) / 15 + (77824 * 3^3 * p2 * p4^2 * p6) / 9 -
(311296 * p3 * p4^2 * p6) / 9 - (77824 * 3^2 * p3 * p4^2 * p6) / 3 -
(77824 * p2 * p3 * p4^2 * p6) / 9 + (252928 * 3 * p6^2) / 9 - (1011712 * 3^3 * p6^2) / 3 -
(10584064 * 3^5 * p6^2) / 135 + (2490368 * 3^7 * p6^2) / 315 - (77824 * 3 * p2 * p6^2) / 9 +
(5292032 * 3^3 * p2 * p6^2) / 27 + (2490368 * 3^5 * p2 * p6^2) / 135 -
(622592 * 3^3 * p2^2 * p6^2) / 27 + (2023424 * p3 * p6^2) / 9 +
(4358144 * 3^2 * p3 * p6^2) / 27 - (4358144 * p2 * p3 * p6^2) / 27 -
(1245184 * 3^2 * p2 * p3 * p6^2) / 27 + (622592 * p2^2 * p3 * p6^2) / 27 -
(622592 * 3^3 * p4 * p6^2) / 27 + (622592 * p3 * p4 * p6^2) / 27;
```

In[5]:=

```
Sys = {F1, F2, F3, F4}
```

Out[5]=

$$\left\{\frac{233892}{7} - \frac{13068\,p2}{7} - \frac{673926\,p2^2}{35} + \frac{43208\,p2^3}{5} - 1056\,p2^4 - \frac{2575056\,p3}{35} + \frac{292776\,p2\,p3}{5} - 13200\,p2^2\,p3 + \frac{1232\,p2^3\,p3}{3} + \frac{352\,p2^4\,p3}{9} + \frac{17248\,p3^2}{15} - 1408\,p2\,p3^2 + \frac{1408\,p2^2\,p3^2}{3} + \frac{11264\,p3^3}{9} - \frac{2816\,p2\,p3^3}{27} + \frac{963996\,p4}{35} - 28512\,p2\,p4 + 6336\,p2^2\,p4 + 2112\,p3\,p4 + 1056\,p2\,p3\,p4 - \frac{704}{3}\,p2^2\,p3\,p4 -\right.$$

$$\frac{2816\,p3^2\,p4}{3}+\frac{112288\,p6}{5}-8448\,p2\,p6-\frac{11264\,p3\,p6}{3}+\frac{2816\,p2\,p3\,p6}{9},$$

$$-\frac{11131947}{175}+\frac{21623004\,p2}{175}+\frac{163878\,p2^2}{5}-\frac{404456\,p2^3}{7}+\frac{80132\,p2^4}{5}-1248\,p2^5+$$
$$\frac{13155948\,p3}{175}-\frac{10557144\,p2\,p3}{35}+\frac{740688\,p2^2\,p3}{5}-\frac{56576\,p2^3\,p3}{3}-\frac{1456\,p2^4\,p3}{9}+$$
$$\frac{416\,p2^5\,p3}{9}+\frac{3888352\,p3^2}{105}-\frac{140608\,p2\,p3^2}{15}-\frac{23296\,p2^2\,p3^2}{3}+\frac{3328\,p2^3\,p3^2}{3}-$$
$$\frac{1664\,p3^3}{9}+\frac{86528\,p2\,p3^3}{27}-\frac{3328\,p2^2\,p3^3}{27}-\frac{242268\,p4}{5}+\frac{766272\,p2\,p4}{7}-\frac{273936\,p2^2\,p4}{5}+$$
$$4992\,p2^3\,p4+\frac{189696\,p3\,p4}{5}-19968\,p2\,p3\,p4+\frac{21632}{3}\,p2^2\,p3\,p4-\frac{1664}{9}\,p2^3\,p3\,p4+$$
$$\frac{26624\,p3^2\,p4}{3}-\frac{6656}{3}\,p2\,p3^2\,p4-\frac{6656\,p3^3\,p4}{27}-\frac{137904\,p4^2}{5}+14976\,p2\,p4^2-$$
$$4160\,p3\,p4^2-\frac{1664}{3}\,p2\,p3\,p4^2-\frac{1388192\,p6}{35}+\frac{365248\,p2\,p6}{5}-9984\,p2^2\,p6+$$
$$\frac{1664\,p3\,p6}{3}-\frac{86528\,p2\,p3\,p6}{9}+\frac{3328}{9}\,p2^2\,p3\,p6-19968\,p4\,p6+\frac{6656\,p3\,p4\,p6}{9},$$

$$\frac{4906657779}{1225}+\frac{4073972004\,p2}{1225}-\frac{744147834\,p2^2}{175}-\frac{1255824\,p2^3}{35}+\frac{34787644\,p2^4}{35}-$$
$$\frac{11564624\,p2^5}{35}+\frac{568616\,p2^6}{15}-1088\,p2^7-\frac{20206105344\,p3}{1225}+\frac{558301896\,p2\,p3}{175}+$$
$$\frac{1536397032\,p2^2\,p3}{175}-\frac{104643296\,p2^3\,p3}{21}+\frac{42158368\,p2^4\,p3}{45}-\frac{474368\,p2^5\,p3}{9}-$$
$$\frac{9248\,p2^6\,p3}{9}+\frac{1088\,p2^7\,p3}{27}+\frac{470012736\,p3^2}{35}-\frac{1516736192\,p2\,p3^2}{105}+\frac{174415104\,p2^2\,p3^2}{35}-$$
$$\frac{25750784\,p2^3\,p3^2}{45}-\frac{4352\,p2^4\,p3^2}{3}+\frac{17408\,p2^5\,p3^2}{9}+\frac{132196352\,p3^3}{45}-\frac{36056320\,p2\,p3^3}{27}+$$
$$\frac{2580736\,p2^2\,p3^3}{27}+\frac{139264\,p2^3\,p3^3}{27}-\frac{8704\,p2^4\,p3^3}{81}+\frac{19514368\,p3^4}{135}+\frac{848223636\,p4}{175}-$$
$$\frac{307103232\,p2\,p4}{175}-\frac{18804720\,p2^2\,p4}{7}+\frac{54909184\,p2^3\,p4}{35}-\frac{1235152\,p2^4\,p4}{5}+$$
$$6528\,p2^5\,p4-\frac{1182889104\,p3\,p4}{175}+\frac{307416576\,p2\,p3\,p4}{35}-\frac{44619968}{15}\,p2^2\,p3\,p4+$$
$$\frac{2308736}{9}\,p2^3\,p3\,p4+\frac{20672}{3}\,p2^4\,p3\,p4-\frac{2176}{9}\,p2^5\,p3\,p4-\frac{46823168\,p3^2\,p4}{15}+$$
$$\frac{22961152}{15}\,p2\,p3^2\,p4-87040\,p2^2\,p3^2\,p4-\frac{34816}{9}\,p2^3\,p3^2\,p4-\frac{7058944\,p3^3\,p4}{27}+$$
$$\frac{69632}{9}\,p2\,p3^3\,p4+\frac{4567152\,p4^2}{7}-\frac{30887232\,p2\,p4^2}{35}+\frac{1669536\,p2^2\,p4^2}{5}-$$
$$13056\,p2^3\,p4^2+\frac{4813312\,p3\,p4^2}{5}-\frac{1823488}{3}\,p2\,p3\,p4^2+28288\,p2^2\,p3\,p4^2+$$
$$\frac{4352}{9}\,p2^3\,p3\,p4^2+156672\,p3^2\,p4^2-\frac{34816\,p3^3\,p4^2}{27}-\frac{721344\,p4^3}{5}+78336\,p2\,p4^3-$$
$$21760\,p3\,p4^3-\frac{8704}{3}\,p2\,p3\,p4^3+\frac{201081984\,p6}{35}+\frac{10260928\,p2\,p6}{7}-\frac{97262848\,p2^2\,p6}{35}+$$
$$\frac{10540544\,p2^3\,p6}{15}-43520\,p2^4\,p6-\frac{1180070912\,p3\,p6}{105}+\frac{267156224\,p2\,p3\,p6}{45}-$$
$$\frac{6497536}{9}\,p2^2\,p3\,p6-\frac{17408}{9}\,p2^3\,p3\,p6+\frac{43520}{27}\,p2^4\,p3\,p6-\frac{37322752\,p3^2\,p6}{45}-$$
$$\frac{139264}{3}\,p2\,p3^2\,p6+\frac{139264}{9}\,p2^2\,p3^2\,p6+\frac{1114112\,p3^3\,p6}{27}-\frac{278528}{81}\,p2\,p3^3\,p6+$$
$$\frac{110196992\,p4\,p6}{35}-\frac{11323904\,p2\,p4\,p6}{5}+208896\,p2^2\,p4\,p6+\frac{7685632\,p3\,p4\,p6}{9}+$$
$$\frac{34816}{3}\,p2\,p3\,p4\,p6-\frac{69632}{9}\,p2^2\,p3\,p4\,p6-\frac{278528}{9}\,p3^2\,p4\,p6-104448\,p4^2\,p6+$$
$$\frac{34816}{9}\,p3\,p4^2\,p6+\frac{30620672\,p6^2}{15}-278528\,p2\,p6^2-\frac{1114112\,p3\,p6^2}{9}+\frac{278528}{27}\,p2\,p3\,p6^2,$$

$$-\frac{18377719863}{875}+\frac{11280453444\,p2}{875}+\frac{171738935346\,p2^2}{6125}-\frac{669146712\,p2^3}{35}-\frac{38145692\,p2^4}{25}+$$

$$\frac{139196432\,p2^5}{35} - \frac{120148552\,p2^6}{105} + \frac{1757728\,p2^7}{15} - 2432\,p2^8 + \frac{17161750836\,p3}{245} -$$
$$\frac{132555487704\,p2\,p3}{1225} + \frac{188936304\,p2^2\,p3}{175} + \frac{23523364352\,p2^3\,p3}{525} - \frac{6527499856\,p2^4\,p3}{315} +$$
$$\frac{158404672\,p2^5\,p3}{45} - \frac{1686592\,p2^6\,p3}{9} - \frac{71744\,p2^7\,p3}{27} + \frac{2432\,p2^8\,p3}{27} - \frac{7405768928\,p3^2}{175} +$$
$$\frac{56363396032\,p2\,p3^2}{525} - \frac{2721400704\,p2^2\,p3^2}{35} + \frac{6992053504\,p2^3\,p3^2}{315} - \frac{105556096\,p2^4\,p3^2}{45} +$$
$$\frac{107008\,p2^5\,p3^2}{9} + \frac{48640\,p2^6\,p3^2}{9} - \frac{11602344832\,p3^3}{1575} + \frac{12593110016\,p2\,p3^3}{945} -$$
$$\frac{562925312\,p2^2\,p3^3}{135} + \frac{2023424\,p2^3\,p3^3}{27} + \frac{3171328\,p2^4\,p3^3}{81} - \frac{19456\,p2^5\,p3^3}{27} + \frac{1403419648\,p3^4}{945} -$$
$$\frac{84672512\,p2\,p3^4}{135} + \frac{622592\,p2^2\,p3^4}{3} - \frac{311296\,p2^3\,p3^4}{27} + \frac{4980736\,p3^5}{27} - \frac{2490368\,p2\,p3^5}{81} +$$
$$\frac{311296\,p2^2\,p3^5}{243} - \frac{110968190292\,p4}{6125} + \frac{7666997952\,p2\,p4}{175} - \frac{1516235568\,p2^2\,p4}{175} -$$
$$\frac{571993024\,p2^3\,p4}{35} + \frac{42137136\,p2^4\,p4}{5} - \frac{6410752\,p2^5\,p4}{5} + 34048\,p2^6\,p4 + \frac{2079611712\,p3\,p4}{175} -$$
$$\frac{12063914112\,p2\,p3\,p4}{175} + \frac{6523078784}{105}\,p2^2\,p3\,p4 - \frac{803972992}{45}\,p2^3\,p3\,p4 + \frac{4265728}{3}\,p2^4\,p3\,p4 +$$
$$\frac{245632}{9}\,p2^5\,p3\,p4 - \frac{34048}{27}\,p2^6\,p3\,p4 + \frac{1058031872\,p3^2\,p4}{105} - \frac{351414272}{21}\,p2\,p3^2\,p4 +$$
$$\frac{16041472}{3}\,p2^2\,p3^2\,p4 + \frac{544768}{9}\,p2^3\,p3^2\,p4 - \frac{330752}{9}\,p2^4\,p3^2\,p4 - \frac{147194368\,p3^3\,p4}{135} +$$
$$\frac{4163584}{9}\,p2\,p3^3\,p4 - \frac{1828864}{9}\,p2^2\,p3^3\,p4 + \frac{389120}{81}\,p2^3\,p3^3\,p4 - \frac{2490368\,p3^4\,p4}{9} +$$
$$\frac{622592}{27}\,p2\,p3^4\,p4 + \frac{80773104\,p4^2}{35} + \frac{353553216\,p2\,p4^2}{35} - \frac{435660576\,p2^2\,p4^2}{35} +$$
$$\frac{18612096\,p2^3\,p4^2}{5} - 145920\,p2^4\,p4^2 - \frac{237164992\,p3\,p4^2}{35} + \frac{117533696}{15}\,p2\,p3\,p4^2 -$$
$$2164480\,p2^2\,p3\,p4^2 - \frac{695552}{9}\,p2^3\,p3\,p4^2 + \frac{48640}{9}\,p2^4\,p3\,p4^2 - \frac{5729792\,p3^2\,p4^2}{15} -$$
$$116736\,p2\,p3^2\,p4^2 + \frac{194560}{3}\,p2^2\,p3^2\,p4^2 + \frac{4046848\,p3^3\,p4^2}{27} - \frac{77824}{27}\,p2\,p3^3\,p4^2 +$$
$$\frac{71788992\,p4^3}{35} - 1488384\,p2\,p4^3 + 175104\,p2^2\,p4^3 + 252928\,p3\,p4^3 + \frac{165376}{3}\,p2\,p3\,p4^3 -$$
$$\frac{19456}{3}\,p2^2\,p3\,p4^3 - \frac{77824\,p3^2\,p4^3}{3} - \frac{1058596704\,p6}{35} + \frac{4259381696\,p2\,p6}{175} +$$
$$\frac{542900224\,p2^2\,p6}{35} - \frac{1349896192\,p2^3\,p6}{105} + \frac{37177984\,p2^4\,p6}{15} - 97280\,p2^5\,p6 +$$
$$\frac{26665681024\,p3\,p6}{525} - \frac{23674829312\,p2\,p3\,p6}{315} + \frac{1183318784}{45}\,p2^2\,p3\,p6 - \frac{19922944}{9}\,p2^3\,p3\,p6 -$$
$$\frac{1206272}{27}\,p2^4\,p3\,p6 + \frac{97280}{27}\,p2^5\,p3\,p6 - \frac{37433344\,p3^2\,p6}{315} - \frac{40546304}{45}\,p2\,p3^2\,p6 -$$
$$\frac{6848512}{9}\,p2^2\,p3^2\,p6 + \frac{622592}{9}\,p2^3\,p3^2\,p6 - \frac{30040064\,p3^3\,p6}{27} + \frac{23035904}{81}\,p2\,p3^3\,p6 -$$
$$\frac{311296}{27}\,p2^2\,p3^3\,p6 - \frac{463218176\,p4\,p6}{35} + \frac{197254656\,p2\,p4\,p6}{7} - \frac{59720192}{5}\,p2^2\,p4\,p6 +$$
$$544768\,p2^3\,p4\,p6 - \frac{57327104\,p3\,p4\,p6}{45} + \frac{2373632}{3}\,p2\,p3\,p4\,p6 + \frac{4708352}{9}\,p2^2\,p3\,p4\,p6 -$$
$$\frac{544768}{27}\,p2^3\,p3\,p4\,p6 + \frac{9961472}{9}\,p3^2\,p4\,p6 - \frac{1245184}{9}\,p2\,p3^2\,p4\,p6 - \frac{622592}{81}\,p3^3\,p4\,p6 +$$
$$\frac{3472896\,p4^2\,p6}{5} + 233472\,p2\,p4^2\,p6 - \frac{2412544}{9}\,p3\,p4^2\,p6 - \frac{77824}{9}\,p2\,p3\,p4^2\,p6 -$$
$$\frac{1132125184\,p6^2}{105} + \frac{146231296\,p2\,p6^2}{15} - 622592\,p2^2\,p6^2 + \frac{15097856\,p3\,p6^2}{9} -$$
$$\frac{15564800}{27}\,p2\,p3\,p6^2 + \frac{622592}{27}\,p2^2\,p3\,p6^2 - 622592\,p4\,p6^2 + \frac{622592}{27}\,p3\,p4\,p6^2\Big\}$$

In[6]:=
```
Var = {p2, p3, p4, p6};
```

```
In[7]:= Timing[
         NSolve[N[Sys, 2 MachinePrecision], Var]]

Out[7]= {1.516 Second,
         {{p2 → 38.600158510963414234229540614 51, p3 → 1.45496323949824612223685335 6722,
           p4 → 670.54672235812110699625275635 8, p6 → 12248.23525056953773103559285783},
          {p2 → 7.69861591676712019466045342439 - 0.99419318343660870432645239632 i,
           p3 → 1.99039835084539894876847263977 3 - 0.04778732351649225831010482079 0 i,
           p4 → 15.47886822752413018045056201335 - 5.16490897272878058104577098149 i,
           p6 → 35.58810635811308562227893153 39 - 20.26720002011312600225817119 79 i},
          {p2 → 7.69861591676712019466045342439 + 0.99419318343660870432645239632 i,
           p3 → 1.99039835084539894876847263977 3 + 0.04778732351649225831010482079 0 i,
           p4 → 15.47886822752413018045056201335 + 5.16490897272878058104577098149 i,
           p6 → 35.58810635811308562227893153 39 + 20.26720002011312600225817119 79 i},
          {p2 → 3.93075464889570291424912244 0836, p3 → 1.34043079226441050982853466 4860,
           p4 → 2.95715301530376721444740831 6560, p6 → 2.54292556423270864188962453 0170},
          {p2 → 3.56038452832268423453118709 7000, p3 → 1.46210376752949515871811506 6627,
           p4 → 2.31514858126839832330221258 0286, p6 → 1.65156292030363159305670025 9225},
          {p2 → 3.27322760511332632875752839 936, p3 → 1.89448728725092719474854184 4363,
           p4 → 2.09625880948501693961051951 8085, p6 → 1.51611550263510135972486860 3022},
          {p2 → 3.36694433682619228958796073 8331, p3 → 2.06969906127709463622298530 2604,
           p4 → 2.26318284136230253791707489 6035, p6 → 1.71830277798523276749703751 0228},
          {p2 → 4.04392411814028462483440159 289 + 0.16460091731511569724187289644 3 i,
           p3 → 2.79889783679083902388146652 031 + 0.01573920578321981426242683462 8 i,
           p4 → 3.00088070381088286712613835 744 + 0.07280162432726966223392321122 6 i,
           p6 → 2.55396425288108317934281683 220 + 0.02989184347250623199314855958 6 i},
          {p2 → 4.04392411814028462483440159 289 - 0.16460091731511569724187289644 3 i,
           p3 → 2.79889783679083902388146652 031 - 0.01573920578321981426242683462 8 i,
           p4 → 3.00088070381088286712613835 744 - 0.07280162432726966223392321122 6 i,
           p6 → 2.55396425288108317934281683 220 - 0.02989184347250623199314855958 6 i},
          {p2 → 3.73167193752493452142817171 3881, p3 → 2.54976334886819161695775488 8026,
           p4 → 2.74995830287324766000044217 1962, p6 → 2.28852296032585958410828055 8883},
          {p2 → 3.85326657306478183615086142 6632, p3 → 2.86986717607156205652914945 9925,
           p4 → 2.95713001601527317294894250 1373, p6 → 2.58845006441854597540231552 5438},
          {p2 → 2.94691985795260694233370278 5607, p3 → 2.05344409736002540491923184 6785,
           p4 → 1.96196944953647847751994988 5061, p6 → 1.53156373595662029826247154 1960},
          {p2 → 3.08498869819135998537040694 0803, p3 → 2.22932316366008051929964183 8219,
           p4 → 2.16932318264635628866561767 0699, p6 → 1.76159944686655568834571427 2261},
          {p2 → 3.54638708716818244311218127 289 + 0.02549776295757122655257366462 2 i,
           p3 → 2.73593468118406346430073034 448 - 0.02021137398273307085323042518 8 i,
           p4 → 2.78091372033730222725503100 775 - 0.02826282101819470975659017811 5 i,
           p6 → 2.42624842426388544309637325 8641 - 0.02924896148144614451171129326 0 i},
          {p2 → 3.54638708716818244311218127 289 - 0.02549776295757122655257366462 2 i,
           p3 → 2.73593468118406346430073034 448 + 0.02021137398273307085323042518 8 i,
           p4 → 2.78091372033730222725503100 775 + 0.02826282101819470975659017811 5 i,
           p6 → 2.42624842426388544309637325 8641 + 0.02924896148144614451171129326 0 i},
          {p2 → 4.06531917518590818524993310 961 - 0.08465153218570210949126907815 0 i,
           p3 → 3.33393740866684166477548461 995 - 0.11460316376346449018568716855 6 i,
           p4 → 3.50114431971106459614225250 310 - 0.13280799411519352879654957356 5 i,
           p6 → 3.20516323535003881259432639 365 - 0.14934626909306407760982945479 0 i},
          {p2 → 4.06531917518590818524993310 961 + 0.08465153218570210949126907815 0 i,
           p3 → 3.33393740866684166477548461 995 + 0.11460316376346449018568716855 6 i,
           p4 → 3.50114431971106459614225250 310 + 0.13280799411519352879654957356 5 i,
           p6 → 3.20516323535003881259432639 365 + 0.14934626909306407760982945479 0 i},
          {p2 → 3.41039955592299986520618021 2852, p3 → 2.63982022724457909614783561 4634,
           p4 → 2.67078739291230652150690647 3997, p6 → 2.30250889910847254813758804 9105},
          {p2 → 2.67074555189383274034966462 9445, p3 → 2.00229453940070839924906698 4236,
           p4 → 1.79658796579434101195290500 1812, p6 → 1.43168883272813740427123045 7420},
```

```
{p2 → 2.784202441310058554824077254490, p3 → 2.172252588137002058011812186923,
  p4 → 1.988936051004880139169051990648, p6 → 1.651544060280237721801656378622},
 {p2 → 1.904413901905022925823872255719, p3 → 1.312163827080700659272418998602,
  p4 → 0.946319357645433001924158118058, p6 → 0.532558348226846717566204208803},
 {p2 → 2.328502190843818239919237610176, p3 → 1.785027357710131653731338381237,
  p4 → 1.499633761650627047406374500205, p6 → 1.145598369794868101735680902531},
 {p2 → 2.026195449153659853757670640535, p3 → 1.482192939158884675370824849798,
  p4 → 1.142813337121778409447039169393, p6 → 0.753363202972735639047038281950},
 {p2 → 2.461052545707877802625767777023, p3 → 1.956329557569852804088369549319,
  p4 → 1.698087924671812827818995832075, p6 → 1.368086790129084242478894782031},
 {p2 → 3.074690766359377076310071570 71 + 0.285142194651457744131264280084 i,
  p3 → 2.635988970097541882455615620 01 + 0.009083202592245910699837441787 i,
  p4 → 2.468580389659578578122180207954 + 0.038227949897923847179019614270 i,
  p6 → 2.242915961685472279391541163087 + 0.009415723608015531543490994049 i},
 {p2 → 3.074690766359377076310071570 71 - 0.285142194651457744131264280084 i,
  p3 → 2.635988970097541882455615620 01 - 0.009083202592245910699837441787 i,
  p4 → 2.468580389659578578122180207954 - 0.038227949897923847179019614270 i,
  p6 → 2.242915961685472279391541163087 - 0.009415723608015531543490994049 i},
 {p2 → 2.920796528488233845747752393860, p3 → 2.485522534748737466303986136156,
  p4 → 2.332266235156253047923622137390, p6 → 2.053236699581373070597056785642}}}
```