**Skalarna in vektorska polja v Mathematici**

**Scalar and vector fields in Mathematica**

**Vili Volčini**

[**vilivolcini@gmail.com**](mailto:vilivolcini@gmail.com)**,** [**vili.volcini@student.fmf.uni-lj.si**](mailto:vili.volcini@student.fmf.uni-lj.si)

**Povzetek**

V tem povzetku bom prestavil glavno idejo skalarnih ter vektorskih polj. Prikazal bom tudi risalne funkcije od Mathematice, s katerimi se izrisujejo skalarna ter vektorska polja. Prikazal bom tudi nekaj osnovnih primerov skalarnih polj ter vektorskih polj z elementarnimi funkcijami.

**Ključne besede**

Skalarno polje, vektorsko polje, Mathematica

**Abstract**

In this seminar work I will show main idea about scalar and vector fields. I also will show drawing functions of Mathematica, which are good for drawing scalar and vector fields. Some basic scalar and vector functions will be shown too.

**Key words**

Scalar field, vector field, Mathematica

**Uvod**

Namen tega prispevka je, da opišem specifičen Mathematice: Razložil bom idejo skalarnih ter vektorskih polj ter opisal par funkcij iz Mathematice, ki izrisujejo prej omenjena polja. Potem bom primerjal Mathematico z podobnimi programi.

**Skalarna in vektorska polja v Mathematici**

Skalarna polja in vektorska polja so pogosto videna v matematiki in fiziki, vsaj se z njimi lahko modelira naravne pojave. Te naravni pojavi so tok vode, smer vetra, tornado, temperaturne slike – vreme, fizični pritisk, zvočni valovi, elektromagnetne sile, elektrostatičen potencial  [[Electric Fields of Point Charges (1)]](http://supermath.info/ElectricFieldsfromPtCharges.pdf) in tako naprej. Skalarna & vektorska polja se pogosto pojavljajo tudi v računalniškem svetu, posebej pri analiziranju ter obdelavi medijev, kot so slika in posnetki, ter seveda računalniški vid.

Mathematica podpira kar nekaj funkcij za izrisovanje skalarnih polj ter vektorskih polj. Najbolj pogosta skalarna polja, ki se izrisujejo, so dvodimenzionalna polja, enako to velja tudi za vektorska polja. Mathematica podpira tudi izris tridimenzionalnih skalarnih in vektorskih polj.

**Navadna funkcija**

Navadna funkcija ima en vhodni argument – skalar ter vrne skalar.

**Skalarno polje**

Skalarna polja imajo en vhodni argument – vektor, ter vrnejo skalar.

– skalarno polje nam priredi skalar v kateri koli točki vektorskega prostora.

Obstajajo tudi alternativne notacije za skalarna polja:

Dvodimenzionalno skalarno polje:

Tridimenzionalno skalarno polje:

**Funkcije za izris skalarnih polj v Mathematici**

Mathematica je kar oborožena s funkcijami za izris dvodimenzionalnih skalarnih polj, ima pa tudi manjšo podporo za tridimenzionalna skalarna polja.

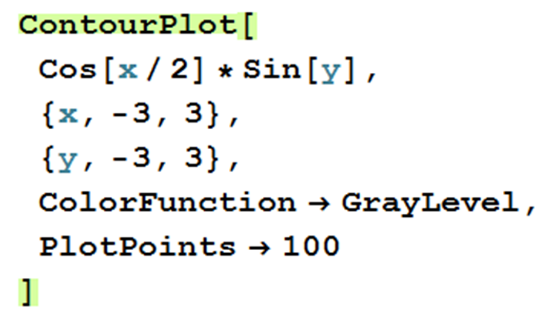
Najbolj pogoste funkcije za izris 2D skalarnih polj so:

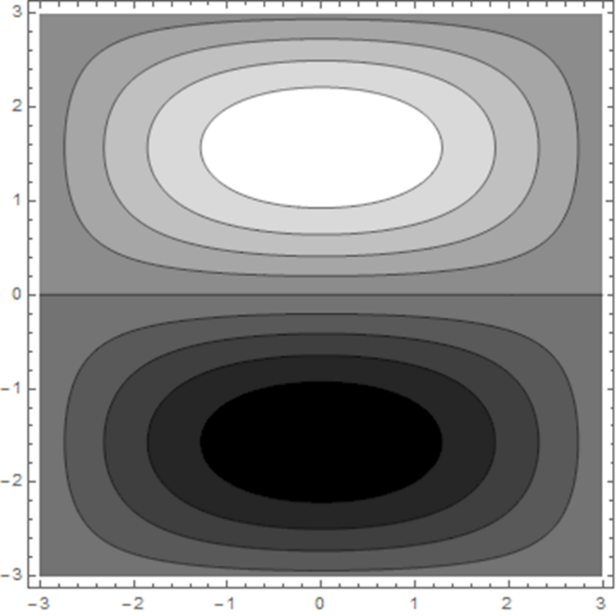
* ContourPlot
* DensityPlot
* DiscretePlot
* Plot3D

**ContourPlot**

Najbolj pogosta funkcija za izris skalarnih polj je ContourPlot. Ta nam izriše graf v obliki reliefa, tako kot vidim pogosto na geografskih zemljevidih (višina gore).

Ta koda nam izriše reliefni graf:

****

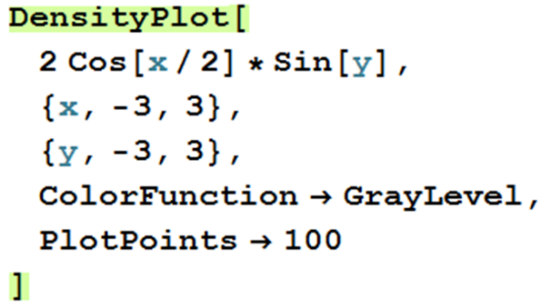
****

Slika 1 ContourPlot

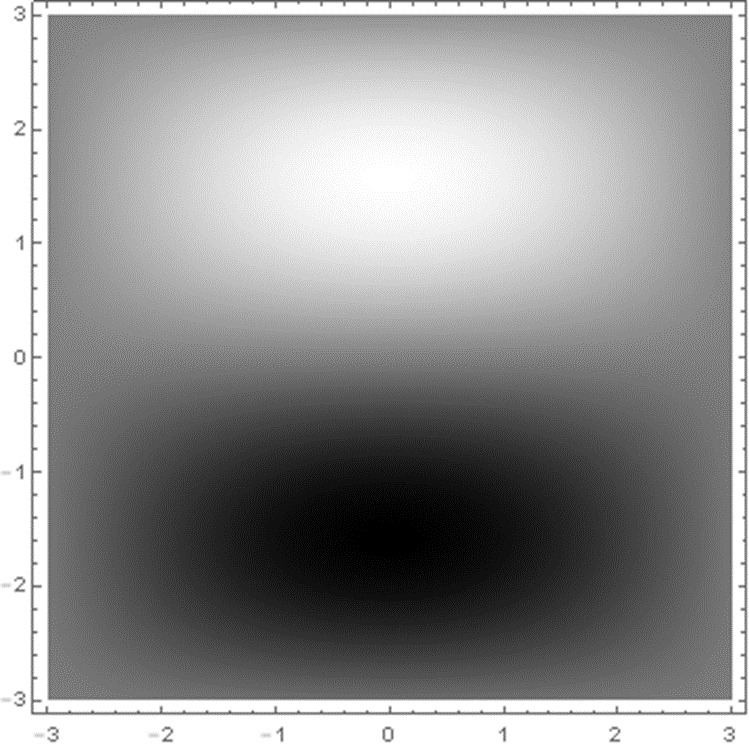
Črte na grafu oziroma »elipsaste« zaprte krivulje ponazarjajo enako vrednost. Bela barva ponazarja večjo vrednost, siva srednjo vrednost, črna pa najnižjo vrednost.

**DensityPlot**

Če pa želimo namesto reliefnega načina izrisati čisti graf oziroma da ima vsak piksel določeno svojo vrednost/barvo, uporabimo DensityPlot. Koda je podobna kot v prejšnjem primeru.



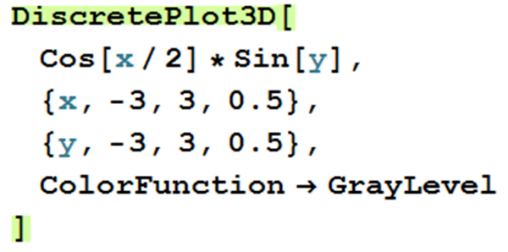
Graf je prikazan na sliki Slika 2. Črna barva predstavlja najnižjo vrednost, bela pa najvišjo.



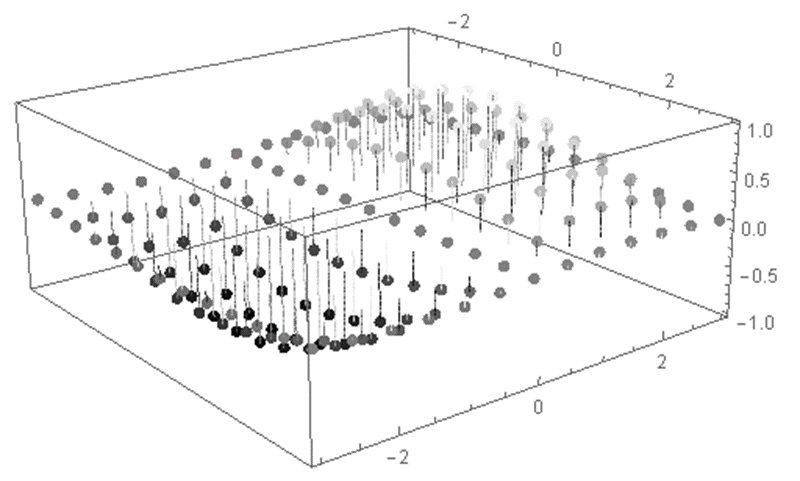
Slika 2 DensityPlot

**DiscretePlot3D**

Skalarna polja lahko izrišemo tudi po kvantiziranih korakih – z enakomernim vzorčenjem.

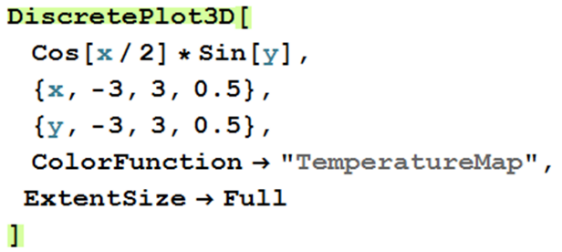


Zgornja koda izriše tak graf, kot je prikazan na Slika 3.

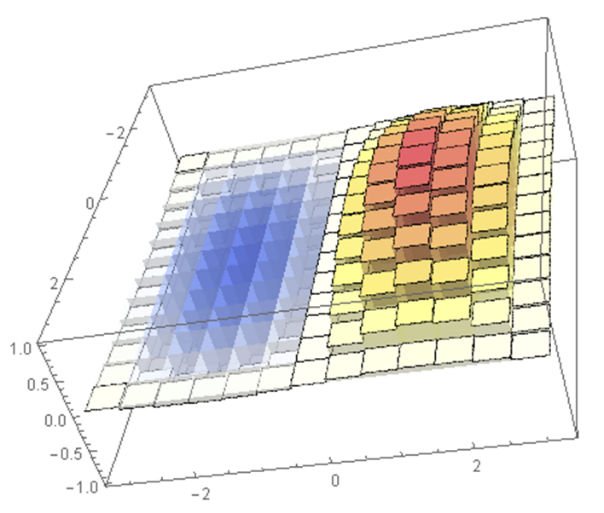


Slika 3 DiscretePlot

Ampak ker je graf na Sliki 3 nerazločen, lahko narišemo boljši graf s to kodo:



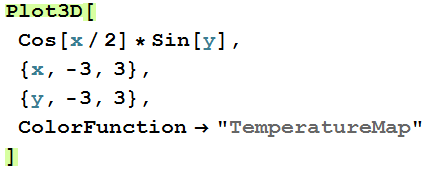
Zgornja koda bolje izriše skalarno polje, a še vedno je nenatančen - nezvezen(Slika 4).

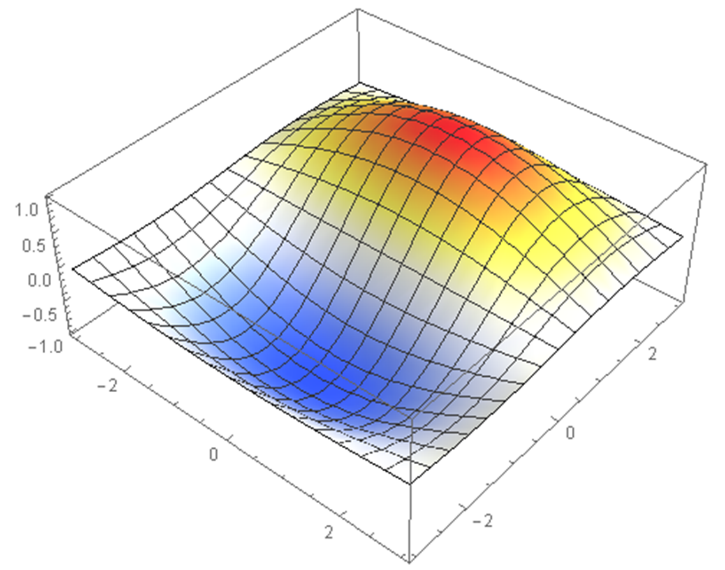


Slika 4 DiscretePlot

**Plot3D**

Še boljši graf se lahko izriše s pomočjo Plot3D. Izriše se čisto gladek graf oziroma funkcija. Naslednja koda generira tak graf, ki je prikazan na Sliki 5.





Slika 5 Plot3D

Plot3D funkcija (graf na Sliki 5) je ena izmed najbolj uporabnih funkcij v Mathematici za izris.

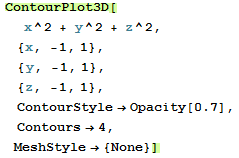
**3D skalarno polje**

Podpora za izris 3D skalarnih polj je v Mathematici malo slabša, toda vedno lahko napišemo svoje funkcije, ki izrisujejo tisto, kar želimo.

Tu bom opisal ContourPlot3D.

**ContourPlot3D**

V naslednji kodi spodaj izrisujem skalarno polje .

****

Slika 6 ContourPlot3D

Graf, izrisan na sliki Slika 6, izrisuje ukrivljene ravnine. Tam, kjer ta ravnina enake barve, tam je povsod enaka vrednost. Recimo rumena barva (krogla) predstavlja vrednost 0.5. Lahko bi se reklo, da rumena krogla predstavlja enačbo, kar je čisto res. Enako velja za ostale barvne sfere (modra, zelena, rdeča) . Vsaka barva predstavlja svojo enačbo oziroma svojo vrednost.

**Vektorsko polje**

Vektorska polja imajo en vhodni argument – vektor, ter vrnejo vektor enake dimenzije.

– vektorsko polje nam priredi vektor v kateri koli točki vektorskega prostora.

Najbolj pogosta vektorska polja so 2D in 3D.

**Funkcije za izris vektorskih polj v Mathematici**

Mathematica ima kar dobro podporo za izris 2D vektorskih polj, za 3D malo manj.

Glavne funkcije za izris 2D in 3D vektorskih polj so:

* VectorPlot
* StreamPlot
* StreamDensityPlot
* VectorPlot3D

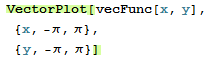
Funkcija za demonstracijo funkcij **VectorPlot**, **StreamPlot** in **StreamDensityPlot** je:



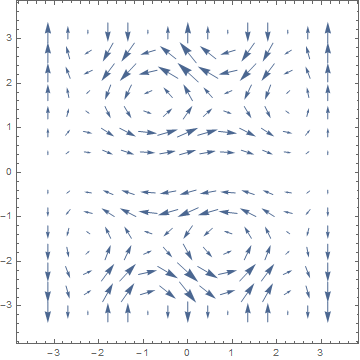
Funkcija za demonstracijo **VectorPlot3D** je . Torej ta funkcija vrne enak vektor, kot smo ga dali notri.

**VectorPlot**

Ker vektorsko polje vrne vektor, se spodobi, da se vektorsko polje vizualizira s puščicami, ki predstavljajo vektorje.



Zgornja koda proizvede tak graf, kot je prikazan spodaj na Sliki 7.

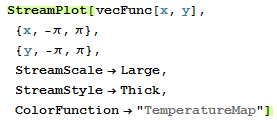


Slika 7 VectorPlot

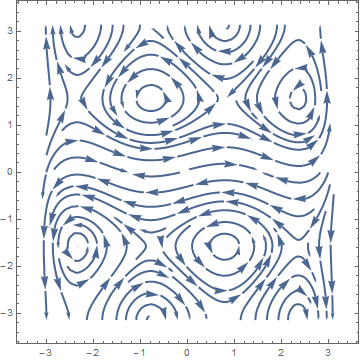
Večja je puščica na grafu, daljši oziroma večji je tam vektor. Take grafe velikokrat vidimo tudi na vremenskih slikah, ki prikazujejo smer in moč vetra.

**StreamPlot**

Včasih naraste želja, da bi predstavljali tokove snovi, kot so recimo morski tokovi.



Zgornja koda izriše tak graf, kot je prikazan spodaj na Sliki 8. Podobne slike vidimo na vremenskih napovedih, ko se vizualizira tokove vetra.

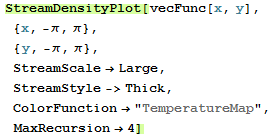


Slika 8 StreamPlot

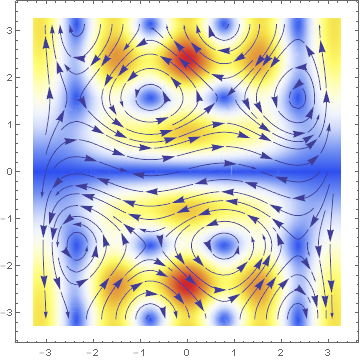
Povezava Plotting a set of trajectories [3] prikazuje tudi, kako je možno izrisovati 3D vektorska polja na tak način, kot je prikazan na Slika 8, ampak v treh dimenzijah.

**StreamDensityPlot**

Če želimo poleg tokov – 'stream' izrisati še moč toka – 'density', je StreamDensityPlot pravi ukaz.



Zgornja koda nariše tako sliko, kot je prikazana na Sliki 9.



Slika 9 StreamDensityPlot

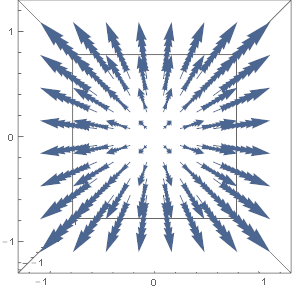
Glavna značilnost tega grafa je, da se poleg vektorskega polja izriše še skalarno polje v ozadju. To skalarno polje predstavlja magnitudo oziroma dolžino vektorjev.

**VectorPlot3D**

S preprostim ukazom VectorPlot3D izrišemo lahko 3D vektorsko polje.



Ko zaženemo zgornjo kodo v Mathematici, se ustvari tak graf, kot je prikazan na Sliki 9.



Slika 10 VectorPlot3D

**Moje mnenje o orodju**

Menim, da je Mathematica na splošno dobro orodje, ki pomaga pri razvoju, in vidim, da bi se lahko uporabljala kjerkoli, kjer je potrebna matematika. Poleg uporabnosti pri razvoju mislim, da je Mathematica dobra tudi za učenje, vsaj lahko notri napišemo formulo in dobimo takojšen odziv – kar je ključnega pomena. Mathematica je tudi fina za igračkanje oziroma experimentiranje.

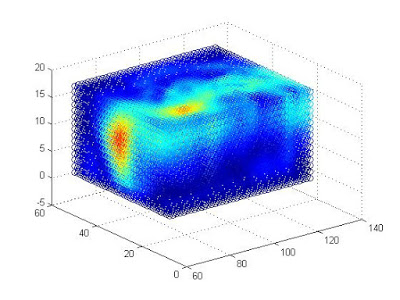
Mathematica podpira tudi veliko drugih grafov, prikazano na viru:

[University of Texas: Functions and Graphing [4]](http://www.cs.utexas.edu/~evanott/PHY110C_Textbook/static/data_analysis/Mathematica/functions_graphs.html)

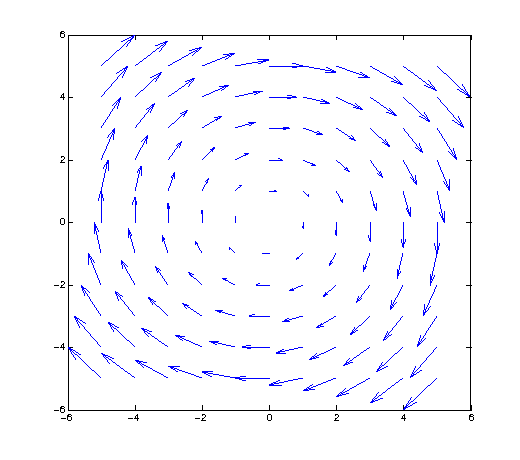
**Primerjava z ostalimi orodji**

*Matlab:*

Je tudi zelo močno orodje, posebej narejen za računanje z matrikami. Je tudi zelo hiter, vsaj se koda tolmači za vsak procesor (CPU) posebej, računa pa numerično. Ima tudi funkcije za izris skalarnih in vektorskih polj, ki se lahko kosajo z grafi od Mathematice (Slika 11 in Slika 12).



Slika 11 Matlab 3D skalarno polje

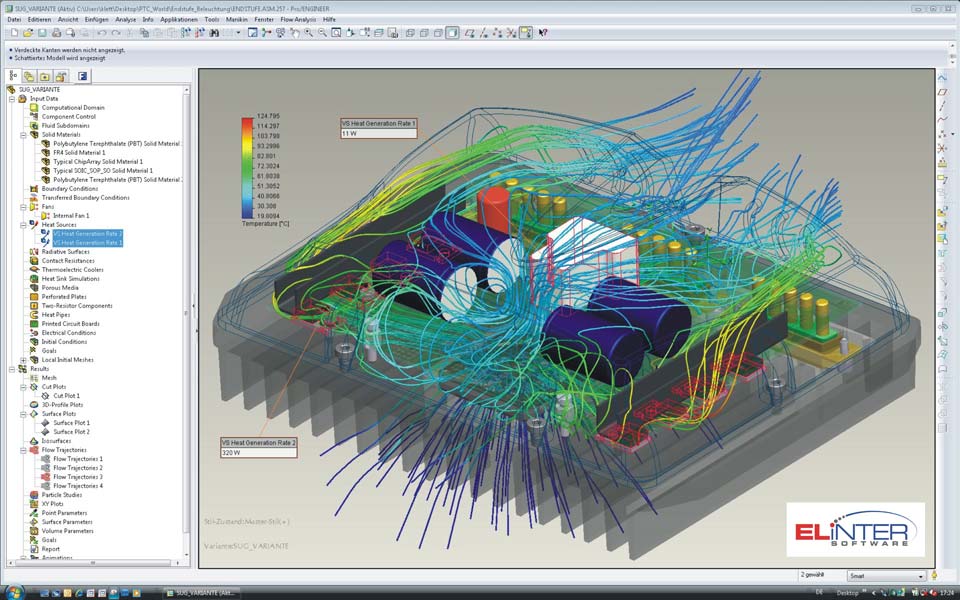


Slika 12 Matlab 2D vektorsko polje

Izvor slik: Google Image Search

*Octave:* Ker je Octave open-source verzija Matlaba, tudi podpira izrise skalarnih ter vektorskih polj.

*FloEFD:* Ta program je bolj namenjen fluidnim simulacijam, pri njem pa so skalarna in vektorska polja ključnega pomena. Z fluidnim simulatorjem lahko simuliramo, kako se veter oz. zrak premika skozi prostor in čas (vektorsko polje), simulira se lahko tudi temperatura (skalarno polje). En primer simulacije iz tega programa je prikazan na 'Slika 13'.



Slika 13 FloEFD

Posnetek na youtube-u, ki prikazuje simulacijo prikazano na 'Slika 13'

<https://www.youtube.com/watch?v=Unn9leNECqg>

Ampak FloEFD je že drugi spekter, izven tematike tega poročila – če želimo samo izrisovati vektorska in skalarna polja, je Mathematica čisto v redu.

**Zaključek**

Mathematica je zares univerzalno orodje, z njim se da marsikaj narediti. Kar se pa tiče skalarnih in vektorskih polj, je tudi Mathematica že kar dobro oborožena s funkcijami za risanjele le teh. Mathematica je tudi za učenje. Sploh, če uporabljamo Manipulate – hitro dobimo občutek, kako se določena funkcija obnaša. Po mojih izkušnjah je dobra tudi za delo – tudi če delamo recimo igrico v C#, zna biti Mathematica zelo uporabna, če želimo na hitro zmodelirati kakšno funkcijo, kajti s pomočjo plot funkcij in Manipulate-a hitro dobimo občutek. (Nato to funkcijo 'prepišemo' v C# kodo itd.).

Za izris polj je Mathematica odlična, saj to vse zmore. Tudi kakšne bolj zahtevne zadeve se da narediti brez težave, recimo problemi iz diferencialnih enačb.

Seveda Mathematica podpira tudi računanje s skalarnimi ter vektorskimi polji, simuliranje v teh poljih (recimo elektromagnetne simulacije), toda stvar se hitro zakomplicira zaradi velike količine kode ter nas lahko stane veliko časa, vsaj Mathematica računa simbolno. Seveda se lahko računa numerično, toda po mojih izkušnjah tudi numerični način ni tako hiter kot kakšen »standalone« program napisan v C++ jeziku.

Toda za kakšne velike/resne simulacije je po mojem mnenju bolje iz nule napisati program v C++ ter nujno paralelizirano na grafičnih procesorjih ali pa za to že temu namenjen program uporabljati.

Če pa želimo samo izrisovati skalarna in vektorska polja, torej kakšne funkcije ali direktno iz podatkov (matrik), je Mathematica odlično orodje za to.

Zanimivo je tudi to, da je nekdo naredil kopijo igre Minecraft:

<http://mathematica.stackexchange.com/questions/19669/mathematica-minecraft>

Najbolj zanimivo mi je, da je Mathematica zastonj za Raspberry Pi, seveda za nekomercionalno rabo:

<http://www.wolfram.com/raspberry-pi/>

**Viri**

Moja seminarska naloga napisana v LaTeXu.

**Spletni Viri**

1. Electric Fields of Point Charges <http://supermath.info/ElectricFieldsfromPtCharges.pdf>
2. FloEFD program za fluidno simulacijo <http://www.mentor.com/products/mechanical/floefd/>
3. Murray Bourne: Vector fields – a simple and painless introduction <http://www.intmath.com/blog/mathematics/vector-fields-a-simple-and-painless-introduction-3345>
4. University of Texas: Functions and Graphing <http://www.cs.utexas.edu/~evanott/PHY110C_Textbook/static/data_analysis/Mathematica/functions_graphs.html>
5. Uporabniško ime: halirutan - Plotting a set of trajectories <http://mathematica.stackexchange.com/questions/2099/plotting-a-set-of-trajectories-not-a-vector-field-in-3d>
6. Vector Fields <http://mathforum.org/mathimages/index.php/Vector_Fields>
7. Weisstein, Eric W. "Scalar Field." From MathWorld--A Wolfram Web Resource. <http://mathworld.wolfram.com/ScalarField.html>
8. Weisstein, Eric W. "Vector Field." From MathWorld--A Wolfram Web Resource. <http://mathworld.wolfram.com/VectorField.html>
9. Wikipedia – skalarna polj <http://en.wikipedia.org/wiki/Scalar_field>
10. Wikipedia – vektorska polja <http://en.wikipedia.org/wiki/Vector_field>
11. Wolfram – referenca za Mathematico <http://reference.wolfram.com/language/>