**Analiziranje funkcij in grafični objekti v EffeDiX-u**

**Functions analisys and graphic objects with EffeDiX**

**Stephanie Furlan**

[**stephanie.furlan@student.fmf.uni-lj.si**](mailto:stephanie.furlan@student.fmf.uni-lj.si)

**Povzetek**

V tem prispevku sem opisala delovanje programa in njegove sposobnosti. Osredotočila sem se predvsem na funkcije, ker se program ukvarja večinoma s temi. Poleg tega pa sem prikazala nekaj primerov in sicer kako različne funkcije definiramo. Nato pa sem še opisala, kako lahko definiramo še celo vrsto drugih grafičnih objektov. Pri spoznavanju programa sem ugotovila, da je izredno uporaben, ker ima celo vrsto možnosti za definiranje različnih funkcij in da je pri uporabi res intuitiven. Mislim, da je program res primeren za uporabo analiziranja funkcij. Ob zaključku prispevka sem na kratko opisala slabe in dobre plati programa, ki sem jih ugotovila pri uporabi in spoznavanju le-tega.

**Ključne besede**

Analiziranje funkcij, Grafični objekti, Tabele

**Abstract**

In this article I wrote about the functions and abilities of the program. I focused on the functions, because this program deals mainly with them. I also showed some examples, which explain how we define functions. I described how we can define other graphic objects too. I arrived at the conclusion that this program is really useful, because it has a lot of possibilities to define various functions and its use is really intuitive. I think, that it is so suitable for analising functions. At the end I shortly described the good and bad sides of it, that I discovered when I was using it.

**Key words**

Functions analysis, Graphic objects, Table

**Uvod**

Podrobneje sem se hotela osredotočiti na funkcije in sicer bom prikazala, kako je z EffeDiX – om preprosto risati celo vrsto funkcij ter opazovati, kako se neka funkcija obnaša. Opisala bom še, kako lahko definiramo še celo vrsto drugih grafičnih objektov.

**Predstavitev orodja**

1. **Značilnosti orodja**

EffeDiX je italijanski program Paola Lazzarinija, ki omogoča risanje številnih grafičnih objektov, kot so grafi elementarnih funkcij, grafi zaporedij, grafi funkcij na različnih intervalih, grafi prvega in drugega odvoda, parametrične krivulje, polarne krivulje, vektorski prostori, integralske krivulje, Riemannov integral in grafi v eni, dveh ali treh dimenzijah. Poleg tega vsebuje vrsto možnosti za definiranje točk, daljic, premic, mnogokotnikov, krožnic, elips, parabol in hiperbol.

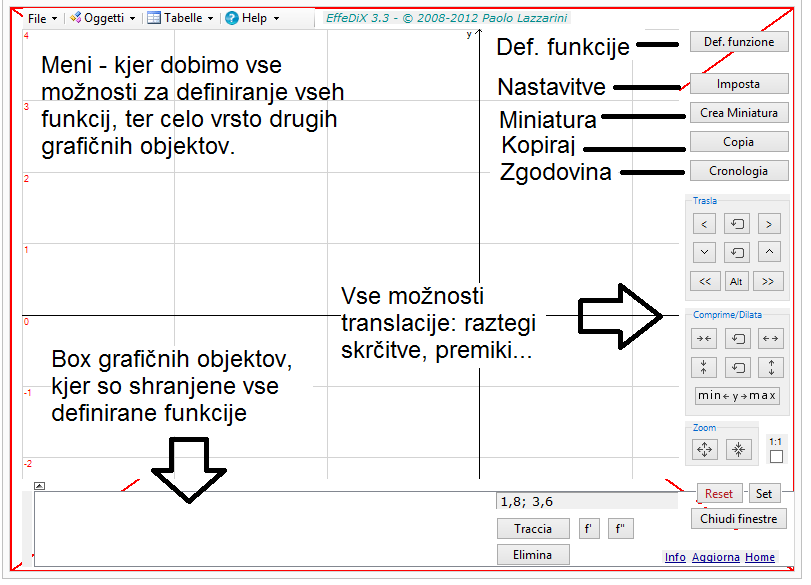
Če se za trenutek zaustavimo pri izvoru imena EffeDiX, lahko opazimo, da izhaja iz italijanščine. Namreč, ''Effe Di X'' pomeni f od x = , kar označuje predpis funkcije. Prav zaradi tega, se program osredotoča predvsem na te.

Program je namenjen študiju funkcij in je pri uporabi zelo intuitiven. Z EffeDiX-om lahko kljub že navedenim možnostim objekte tanzliramo, raztegnemo, skrčimo in povečamo enostavno s pomočjo miške ter ugrajenih ukazov. Obnašanje funkcij in drugih objektov lahko na enostaven način opazujemo z definiranjem drsnikov. Z uporabo teh je mogoče graf funkcije animirati in ob tem razumeti, kako se ta obnaša glede na določene parametre.

V novi različici Effedix-a je mogoče vklopiti rendering 3D za vsak graf funkcije in shraniti delo na zaslonu v obliki miniature.

Program je dostopen vsem, vendar ga dobimo samo v italijanski različici. Dobimo ga lahko na naslednji spletni strani: <http://www.paololazzarini.it/EffeDiX/> . Na naslednji spletni strani dobimo tudi krajši priročnik <http://users.libero.it/prof.lazzarini/EffeDiX/Guida.pdf>, ki nam pomaga pri uporabi samega programa na začetku.

1. **Zaslon**

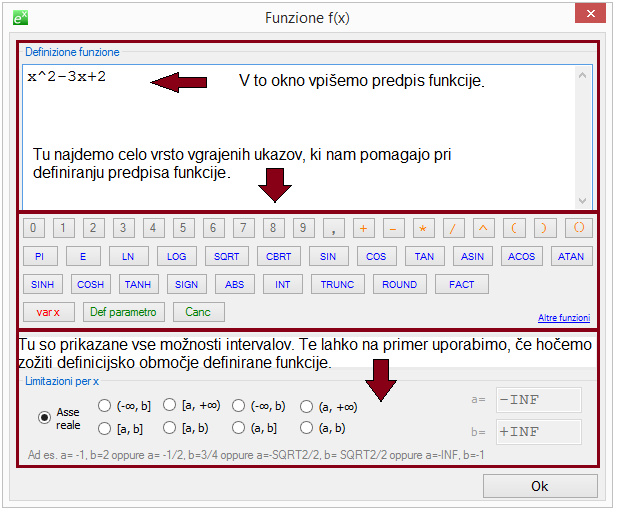
Na spodnji sliki (''Slika 1: Zaslon'') je prikazana tipična slika zaslona, taka, kot se prikaže, ko zaženemo program.

Slika 1: Zaslon

1. **Funkcije**

EffeDiX nam omogoča definiranje grafičnih objektov, kot so tudi funkcije. Lahko definiramo grafe elementarnih funkcij, kot so linearne funkcije, kvadratne funkcije, potenčne funkcije, polinomske funkcije, racionalne funkcije, logaritemske funkcije, trigonometrične funkcije in druge, kot so funkcijska zaporedja, grafi zaporedij delnih vsot ter graf prvega in drugega odvoda.

1. **Definiranje funkcij**

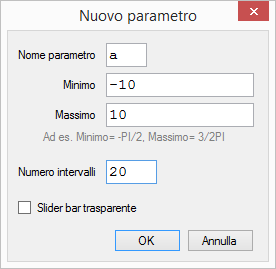
Definiranje funkcije in izris grafa funkcije je s samim programom zelo enostavno. Funkcijo lahko definiramo na dva načina. Najbolj enostaven način je ta, da z miško kliknemo na *Def. Funzione* (*Predpis funkcije*) v zgornjem desnem kotu zaslona. Lahko pa tudi kliknemo na *Oggetti* (*Predmeti*) v Menu-ju in nato izberemo *Grafico di una funzione* (*Graf funkcije* ). V obeh primerih se na zaslonu prikaže novo okno (kot prikazano v spodnji sliki “Slika 2: Definiranje funkcij”) . V polje *Definizione funzione* (*Predpis funkcije*) napišemo predpis funkcije, ki jo hočemo obravnavati. EffeDiX ima že vgrajene ukaze, ki nam pomagajo pri zapisu predpisa funkcije. In sicer lahko predpis le-te definiramo tako, da kliknemo na že vgrajene ukaze, ki so prikazani pod vnosnim oknom *Definizione funzione* (*Predpis funkcije*). Potrebno je paziti na zapis predpisa. Važno je, da je pri predpisu spremenljivka napisana z malimi črkami, funkcijo pa lahko napišemo bodisi z velikimi kot z malimi črkami. Pod vnosnim oknom imamo še možnost, da določimo za katere vrednosti bo funkcija narisana. Če pa tega ne definiramo, bo EffeDiX pri risanju grafa funkcije že upošteval, kje je funkcija definirana. Ko je predpis funkcije definiran, kliknemo na *OK*. Na zaslonu se v spodnjem oknu, ki ga imenujemo *Box grafičnih objektov* (*Box degli oggetti grafici*), prikaže definirana funkcija. Če jo hočemo narisati, je potrebno z miško klikniti na *Traccia* (*Izriši*) ali pa dvakrat na predpis definirane funkcije.

Slika 2: Definiranje funkcij

1. **Elementarne funkcije**

Elementarne funkcije definiramo po že opisanem postopku v prejšnjem sklopu. Ker lahko definiramo celo vrsto funkcij, sem se odločila, da bom prikazala postopek definiranja dveh elementarnih funkij in sicer ter, kjer bo nek parameter. S pomočjo parametra bomo lahko pogledali, kako se funkcija spreminja in sicer kako se ta obnaša, če spreminjamo ta parameter. Izbrala sem si ta primer, ker sem hotela poudariti dejstvo, da ima tudi program EffeDiX možnost drsnikov, kot na primer GeoGebra.

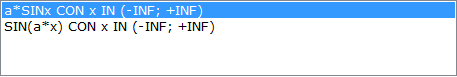
Najprej je potrebno definirati parameter . To storimo tako, da z miško kliknemo na *Def. Funzione* (*Def. Funkcijo*). Odpre se nam novo okno ter, v spodnjem kotu levo, možnost definiranja novega parametra. Za definiranje parametra zadostuje, da kliknemo na gumb *Def parametro* (*Def parameter*). Odpre se nam novo okno, v katero vpišemo minimum in maksimum parametra (kot prikazano na spodnjih slikah “Slika 3: Nov parameter” in ‘’Slika 4: Parameter a’’) ter nato kliknemo *OK*.



Slika 4: Parameter a

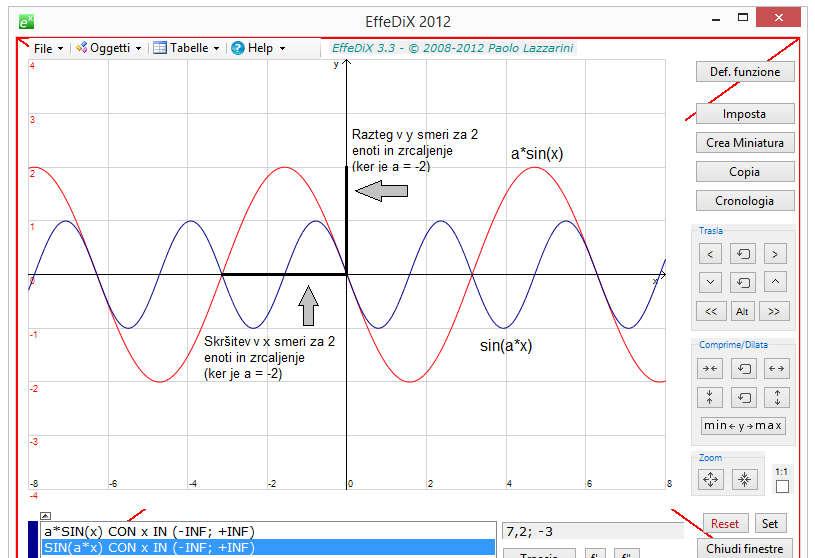
Slika 3: Nov parameter

Nato v okno *Def funzione* (*Def funkcije*) vpišemo predpis prve funkcije in sicer ter kliknemo na *OK*. Isti postopek uporabimo za definiranje funkcije . Komaj definirani funkciji se nam prikažeta v *Box degli oggetti grafici* (*Boxu grafičnih objektov*) (kot prikazano v spodnji sliki “Slika 5:Box”).



Slika 5: Box

Nato kliknemo na *Traccia* (*Izriši*) in se nam obe funkciji izrišeta na zaslon. S premikanjem drsnika lahko gledamo obnašanje obeh funkcij v odvisnosti parametra . V spodnji sliki (“Slika 6: odvisnost a, a = -2”) vidimo, da se funkcija pri vrednosti obrne in raztegne v ordinatni smeri za dve enoti, ter da se funkcija skrči in obrne v abscisni smeri za dve enoti. Pri vrednosti parametra pa se funkciji samo raztegneta v smeri in smeri za dve enoti.



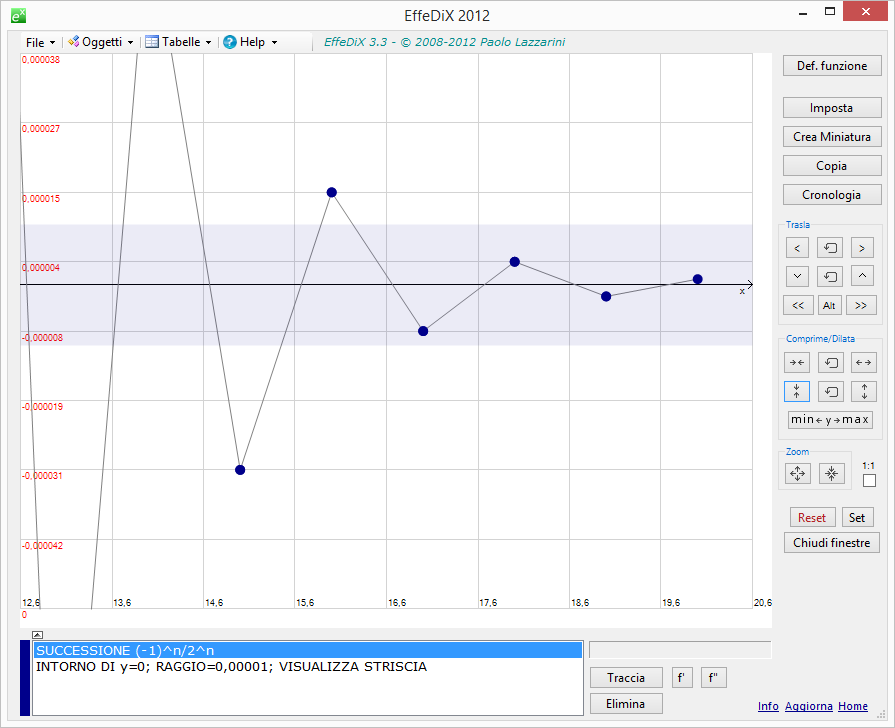
Slika 6: Odvisnost a = -2

1. **Funkcijska zaporedja**

EffeDiX daje možnost, poleg elementarnih funkcij, definiranja funkcijskih zaporedij.

Če hočemo definirati neko funkcijsko zaporedje, izberemo *Menu* *Oggetti* (*Predmeti*) in kliknemo na ukaz *Grafico di una funzione* (*Graf funcije* ), kjer za spremenljivko uporabljamo črko . V vnosno okno *Definizione successione* (*Definicija zaporedja*) vtipkamo predpis funkcijskega zaporedja. Program nam nudi možnost, da lahko narišemo samo vrednosti funkcijskega zaporedja, in sicer tako, da kliknemo na *Traccia solo i punti* (*Nariši samo točke*). Lahko pa si izberemo tudi samo možnost izrisa grafa.

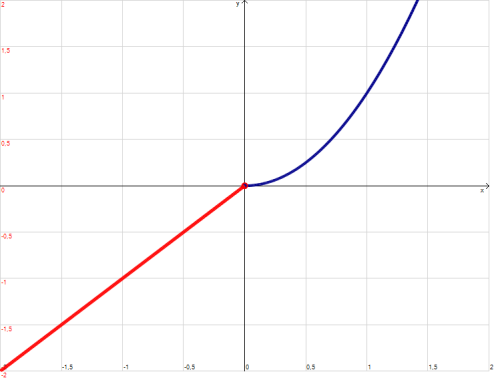
Prikazala bom kako lahko na enostaven način pogledamo, če neko zaporedje konvergira. Izbrala sem si primer alternirajočega zaporedija s predpisom . Zaporedje pa konvergira k številu . Ker ima program možnost definiranja okolice neke točke, bomo najprej okolico točke definirali s pomočjo ukaza *Intorno di un punto* (*Okolica točke*), ki ga najdemo na *Meni-ju* *Predmeti* (*Menu Oggetti*). Nato pa še določimo, da se okolica točke izriše. Potrebno je nato definirati zaporedje. To storimo tako, da v Meni-ju gremo na *Definizione successione* (*Predpis zaporedja*), kjer vpišemo le njen predpis. Nato komaj definirano zaporedje narišemo in animiramo graf v neskončnost. Ob opazovanju grafa vidimo, da so členi zaporedja res v okolici števila , od nekega -ja dalje. Zato zaporedje konvergira k številu (kot prikazano v spodnji sliki ''Slika 7: Konvergenca'').



Slika 7: Konvergenca

1. **Grafi na različnih intervalih**

S programom lahko definiramo tudi grafe funkcij na različnih intervalih. Recimo, da bi hoteli narisati funkciji na intervalu in na intervalu ; dovolj je, da pri definiranju funkcije podamo interval, kje naj bo funkcija narisana. Primer takega grafa je prikazan na spodnji sliki (“Slika 8: Različni intervali”).



Slika 8: Različni intervali

1. **Grafi prvega in drugega odvoda**

Pri definiranju funkcij, nam program nudi možnost prvega in drugega odvoda: na zaslonu obstajata gumba, ki sama izračunata prvi ter drugi odvod že definirane funkcije v *Boxu grafičnih objektov*. Dovolj je, da kliknemo na funkcijo, ki ji želimo izračunati odvod, nato na gumb za izračun odvoda.

Poleg grafa odvoda lahko krivulji narišemo tudi tangente na graf. To storimo tako, da gremo na *Menu* *Tabella* (*Meni Tabela*) in označimo tabelo oblike , ki nam vse vrednosti odvoda za določen izračuna. Vrstice se v tabeli obarvajo glede na to, če je vrednost -a pri odvodu ali .

1. **Tabele**

S pomočjo programa EffeDiX lahko definiramo celo vrsto tabel. To storimo tako, da v *Menu Tabelle* (*Menuju Tabele*) izberemo tabelo, ki jo želimo uporabiti in definirati. V *Menuju Tabele* se nam prikažejo naslednje možnosti:

🡪Tabela (x,f(x))

Tabela prikazuje rezultate funkcije pri različnih vrednostih spremenljivke . Pod *Funzione tabulata* napišemo predpis funkcije. *Intervallo di variazioni per x* pa nas vpraša za interval in sicer od kje do kam bi radi izračunali vrednosti -a. Vpišemo zato levo in desno mejo intervala in korak, s katerim bomo analizirali funkcijo. Barvanje vrstic nam pomaga pri obraunavanju naraščanja in padanja funkcije:

* Če se vrstica v tabeli obarva v *rdeče*, pomeni, da je za tisto vrednost funkcija padajoča.
* Če se vrstica v tabeli obarva v *modro*, pomeni, da je za tisto vrednost funkcija naraščajoča.
* Če vrstica ni obarvana pomeni, da je tam odvod funkcije konstanta.
* V primeru, da je za neko vrednost funkcija nedefinirana, se v tabeli prikaže izraz *Non. Def.*

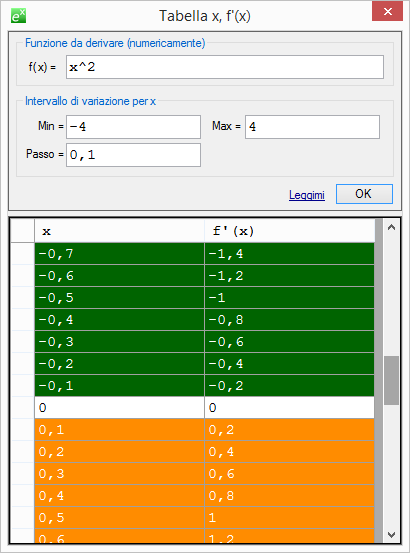
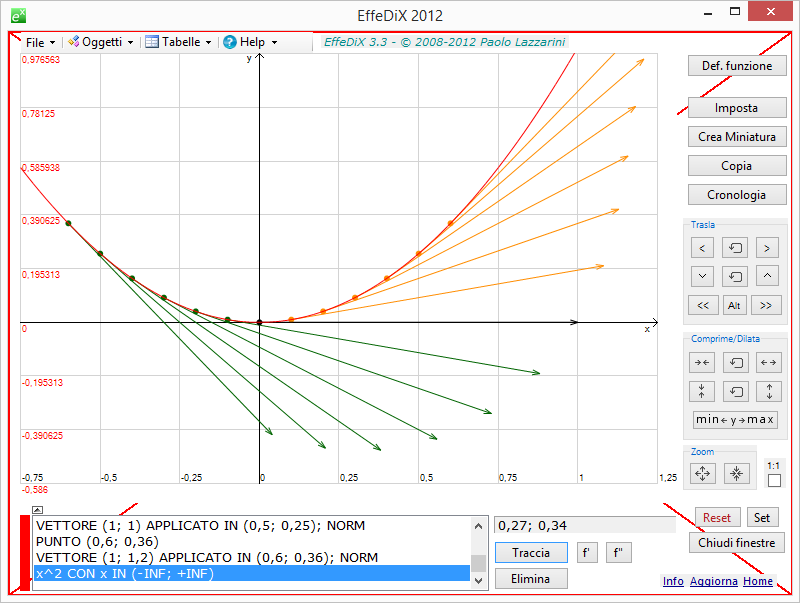
🡪Tabela (x, f’(x))

Tabela prikazuje vrednosti odvoda funkcije pri različnih vrednostih spremenljivke . Vsaka barva vrstice ima točno določen pomen in sicer:

* Če se vrstica obarva v *oranžno*, pomeni, da je odvod v tisti točki pozitiven, torej je funkcija naraščajoča.
* Če se vrstica obarva v *zeleno*, pomeni, da je odvod v tisti točki negativen, torej je funkcija padajoča.
* Če se vrstica ne obarva, je tam , kar pomeni, da je tista točka stacionarna točka funkcije.

Če kliknemo dvakrat na neko vrstico, se bo na grafu funkcije prikazala tangenta, ki predstavlja odvod funkcije v tisti točki.

Na spodnjih dveh slikah(‘‘Slika 9: Tabela(x,f‘(x))‘‘ in ‚‘‘Slika 10 – Tangente na graf‘‘) je prikazano, kako se vrstice tabele obarvajo v različne barve glede na naraščanje in padanje funkcije ter kako zgledajo tangente na nekem grafu.



Slika 10: Tangente na graf

Slika 9:Tabela (x,f'(x))

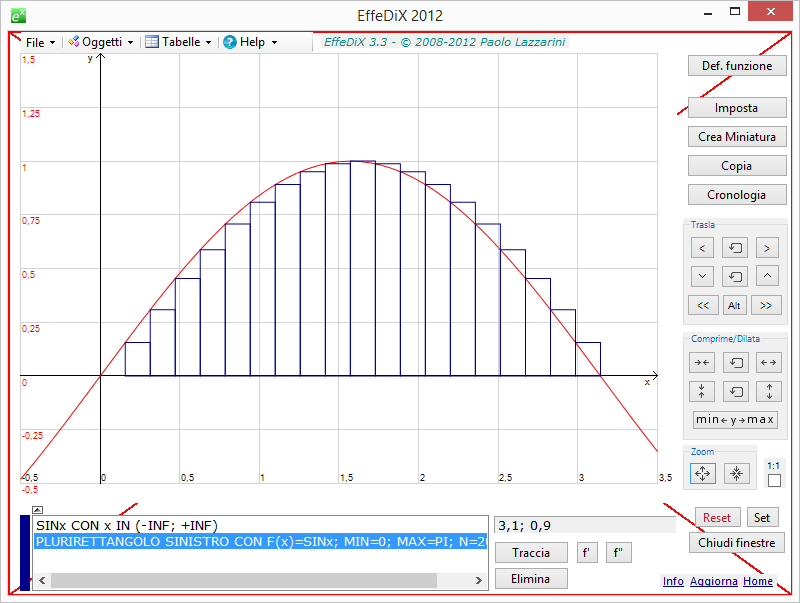
🡪Tabela Zaporedja delnih vsot

Lahko definiramo zaporedje delnih vsot tudi s pomočjo tabele delnih vsot in sicer tako, da vnesemo v vnosno polje predpis vsote za spremenljivko . S tabelo ne moremo grafa funkcije narisati, lahko pa samo narišemo točke tako, da kliknemo dvakrat na vrstico, ki jo želimo narisati.

Če pa hočemo funkcijo narisati, jo moramo definirati. To naredimo s pomočjo ukaza na *Menu Oggetti* (*Meniju Predmeti*) *Grafico somme parziali* (*Graf zaporedja delnih vsot*).

🡪Tabela Riemmanovih vsot

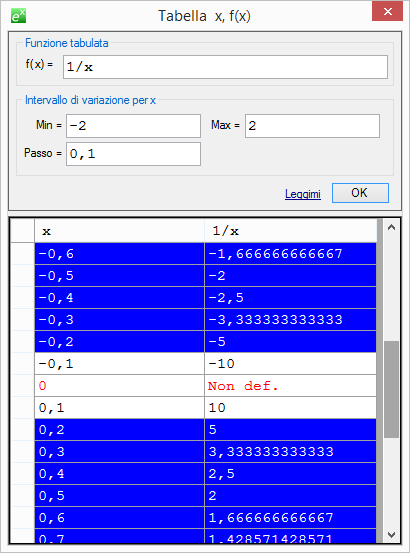
Tabela nam vrača vrednosti Riemmanovih vsot funkcije . V vnosnem oknu, ki se prikaže ko kliknemo na ukaz *Tabela Riemmanovih vsot*, vnesemo predpis funkcije in interval te vsote. Če hočemo na funkciji prikazati pravokotnike, ki predstavljajo Riemmanovo vsoto, gremo na *Menu Oggetti* (*Predmeti*) in kliknemo na ukaz *Plurirettangoli* (*Pravokotniki*) ter napišemo interval, kjer hočemo da se ti pravokotniki prikažejo ter število pravokotnikov na tem intervalu. V spodnji sliki (‘‘Slika 11:Riemmanove vsote‘‘) je prikazano, kako zgledajo grafično Riemmanove vsote funkcije sin(x).



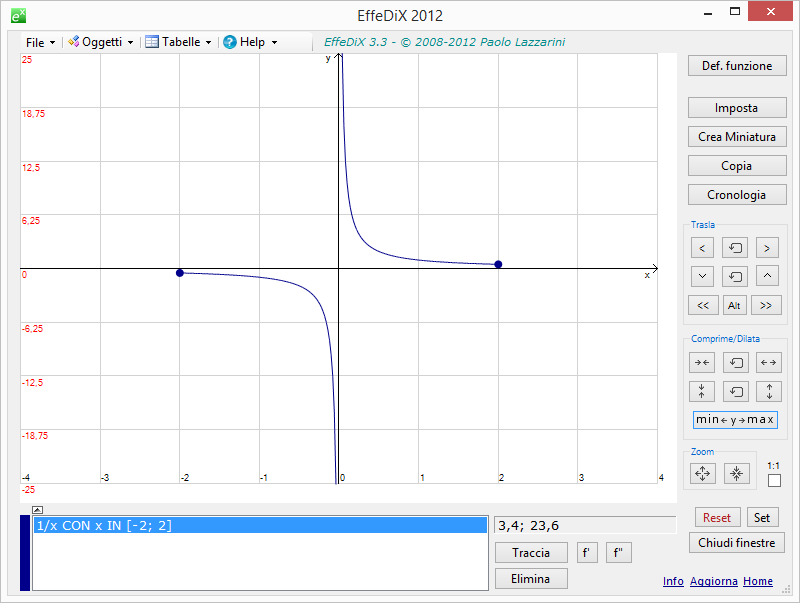
Slika 11: Riemmanove vsote

Primer:

Iščemo tabelo za . Opazimo, da je funkcija nedefinirana v ničli, saj ima tam navpično asimptoto (pol). Ker so vse druge vrednosti pobarvane v modro, pomeni, da je na intervalu funkcija padajoča. (Glej spodnji sliki : „Slika 12: Tabela (x,f(x))“ in „ Slika 13: Funkcija 1/x “)



Slika 12: Tabela (x,f(x))



Slika 13: Funkcija 1/x

1. **Krivulje**

Program EffeDiX nam omogoča tudi risanje različnih vrst krivulj, kot so parametrično podane krivulje, polarne krivulje in integralske krivulje.

**Parametrično podane krivulje**

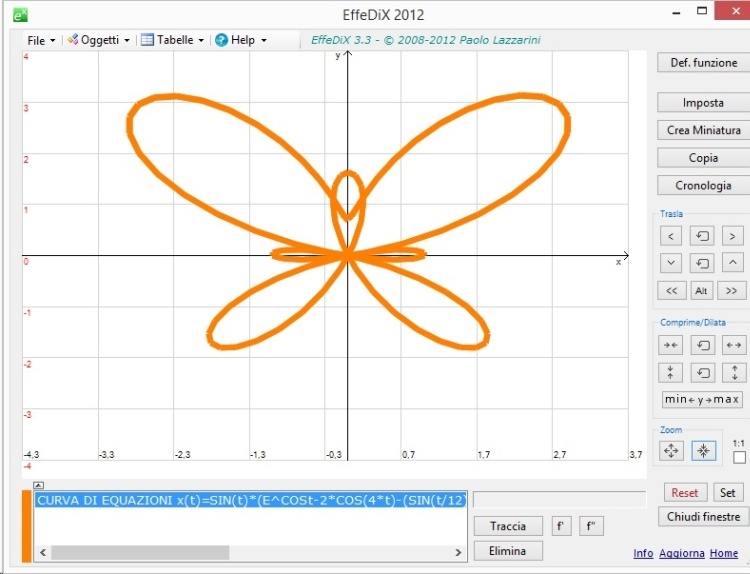
Na zelo enostaven način lahko definiramo neko parametrično krivuljo; to storimo tako, da gremo na *Menu Oggetti* (*Predmeti*) in izberemo ukaz *Curva parametrica* (*Parametrična krivulja*). Nato se na zaslonu odpre novo okno, kamor vpišemo predpisa za in za spremenljivko . Določimo še minimum in maksimum -ja.

**Polarno podane krivulje**

Krivuljo lahko definiramo tudi s pomočjo ukaza *Curva* *in forma polare* (*Polarna* *krivulja*), ki ga dobimo pod *Menu Oggetti* (*Menijem Predmeti*). Enostavno zapišemo polarno enačbo krivulje glede na spremenljivko in tudi v tem primeru določimo minimum in maksimum spremenljivke.

**Integralske krivulje**

Integralsko krivuljo definiramo tako, da gremo na *Menu Oggetti* (*Menu Predmeti*) in kliknemo na *Curva integrale* (*Integralska krivuja*).

V bistvu je definiranje in risanje krivulji z EffeDiX – om res enostavno, dovolj je, da le določimo predpis. Na spodnji sliki (‘Slika 14: Metuljček’’) je prikaz metuljčka, definiranega in narisanega s pomočjo ukaza Parametrično podana krivulja.

Slika 14: Metuljček

1. **Drugi geometrijski objekti**

S programom lahko, poleg funkcij, narišemo še celo vrsto osnovnih geometrijskih elementov. Vse elemente definiramo tako, da gremo na *Menu Oggetti* (*Menu Predmeti*).

TOČKA

Točko v EffeDiX– u definiramo tako, da izberemo z miško *Menu Oggetti* (*Menu Predmeti*) in kliknemo na ukaz *Punto -> Punto(x, y)* (*Točka -> Točka(x, y)*). Prikaže se nam novo okno, v katero vtipkamo koordinato in koordinato *y* točke, ki jo hočemo definirati.

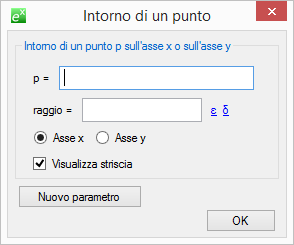
Pod ukazom *Punto* pa imamo še vrsto drugih ukazov:

* *Točka, ki je presek dveh funkcij* (*Punto di intersezione di due grafici*) – tu se nam odpre novo okno. Da dobimo točko, ki je presek dveh grafov, moramo v polja in vpisati za kateri dve funkciji iščemo presek ter na katerem intervalu iščemo točko. Ko kliknemo na *OK*, se nam prikaže na zaslonu samo točka, ne pa grafa definiranih funkcij.
* *Točka, ki je presek funkcije in krožnice* (*Intersezione circonferenza - grafico di funzione*) – kot v prejšnji točki moramo definirati krožnico in funkcijo, za katere iščemo presek. Paziti je treba, da funkciji imata res presek. Če ga nimata, vrne program opozorilo.
* *Točka, ki je minimum funkcije* (*Punto di minimo*) – v tem primeru iščemo točko, ki je minimum definirane funkcije na določenem intervalu.
* *Točka, ki je maksimum funkcije* (*Punto di massimo*) – v tem primeru pa točko, ki predstavlja maksimum funkcije.

Pri naštetih ukazih je treba funkcije ponovno definirati. Ukaz nam pa vrne točko, ki zadošča pogojem ukaza.

OKOLICA TOČKE

V EffeDiX-u obstaja možnost, da definiramo okolico neke točke tako, da gremo na *Menu Oggeti* (*Menu Predmeti*) in kliknemo na *Intorno di un punto* (*Okolica točke*). Nato izberemo točko, kateri hočemo definirati okolico in velikost slednje . Ta ukaz nam je v veliko pomoč pri opazovanju in konvergenci zaporedij.

Velikost predstavlja velikost okolice (npr. Epsilon). Okolico neke točke lahko narišemo bodisi na kot na osi, dovolj je, da označimo *Asse x* (*Os x*) ali *Asse y* (*Os y*). Če vklopimo *Visualizza striscia* (*Prikaži okolico*) se nam bo okolica tiste točke na koordinatnem sistemu obarvala. (Glej spodnjo sliko „Slika 15: Okolica točke“)

**Slika 15: Okolica točke**

DALJICA

Daljico definiramo tako, da izberemo *Menu Oggetti* (*Menu Predmeti*) in pritisnemo na ukaz *Segmento* (*Daljica*). Na zaslonu se prikaže novo okno, ki sprašuje za dva podatka in sicer koordinati začetne in končne točke daljice.

VEKTOR

*Pod ukazom Vettore (Vektor) v Menu* *Oggeti* (*Menuju Predmeti*) imamo tri možnosti in sicer:

* *Vektor določen z začetno točko ter komponentami* (*Date le componenti e il punto di applicazione*) – odpre se nam novo okno v katerega vtipkamo koordinate začetne in končne točke vektorja.
* *Projekcija vektorja u na vektor v* (*Vettore* *p proiezione del vettore u sul vettore v*) – odpre se nam novo okno. Vtipkati moramo koordinate vektorja *u* in vektorja *v* ter koordinate vektorja *p* (te morajo biti na vektorju *p*).
* *Projekcija vektorja u na vektor v, ki je ortogonalen premici q* (*Vettore q proiezione di u su una retta ortogonale a v)* – s tem ukazom dobimo projekcijo nekega vektorja na drug vektor in sicer z zahtevo, da je vektor, na katerega projekciramo, pravokoten na neko drugo premico.

PREMICA

Premico definiramo s pomočjo ukaza *Retta* (*Premica*) na *Menu Oggetti* (*Menuju Predmeti*), kjer se nam prikažejo naslednje možnosti:

* *Navpična premica* (*Verticale*) – odpre se nam novo okno. Prikaže se nam samo možnost premice, ki je navpična, to je vzporedna osi, zato vtipkamo samo koordinato .
* *Premica obike* (*Di equazione* ) – v novo okno vnesemo le smerni koeficient in prosti člen.
* *Premica oblike* (*Di equazione* ) – v novo okno vnesemo *a*, *b*, in *c* premice, ki jo hočemo definirati.
* *Premica skozi dve točki* (*Per due punti*) – podamo točki, čez katere hočemo, da bo premica šla skozi.
* *Premica čez eno točko z danim smernim koeficientom* (*Di data pendenza passante per un punto*) – napišemo koordinati točke, čez katero bo premica šla in smerni koeficient le-te.

MNOGOKOTNIK

S pomočjo ukaza *Mnogokotnik* (*Poligono*) pod *Menu Oggetti* (*Menujem Predmeti*) lahko na preprost način narišemo trikotnik, štirikotnik in pravilni -kotnik. Za definiranje trikotnika podamo tri oglišča, za definiranje štirikotnika podamo štiri oglišča in za definiranje-kotnika podamo dve oglišči ter število stranic.

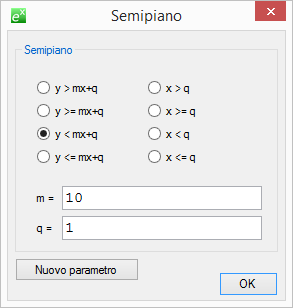
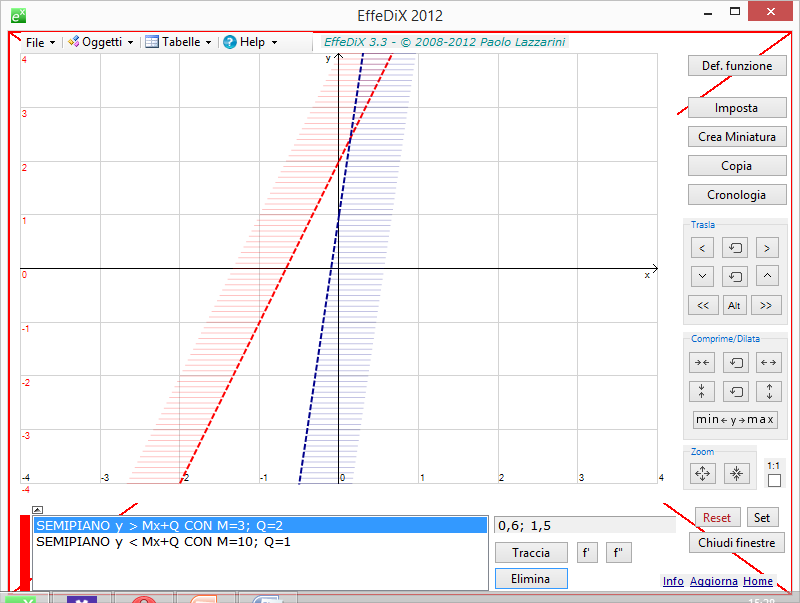
DEL RAVNINE

EffeDiX ima tudi možnost oblikovanja ravnine in sicer tako, da lahko z ukazom *Semipiano* (*Del ravnine*) določimo (in eventuelno zbrišemo) tisti del ravnine, ki nas ne zanima.

Primer:

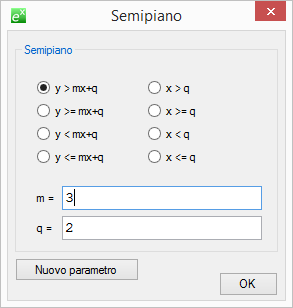
Recimo, da hočemo izbrisati del ravnine za ter .

Prikažeta se dve okni („Slika 17: Del ravnine“ in „Slika 16 : Del ravnine“) v katere vtipkamo željene podatke. Na zaslonu se nam izriše spodnja slika („Slika 17: Izbris“), kjer imata premici enačbi in. Črtkane črte predstavljajo izbrisano ravnino.

****

Slika 17: Izbris

Slika 16: Del ravnine

****

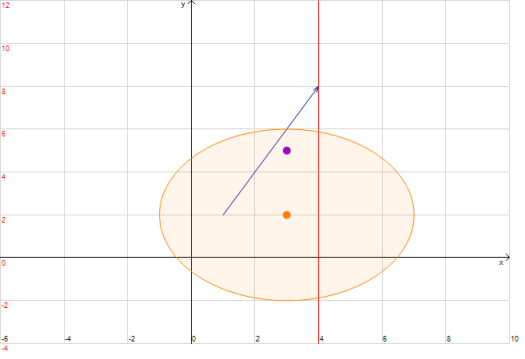
Slika 18: Del ravnine

STOŽNICE

Na zelo enostaven način lahko definiramo in narišemo tudi neko stožnico. Ko izberemo *Menu Oggetti* (*Menu Predmeti*), vidimo, da se nam prikažejo 4 možnosti :

* Krožnica (*Circonferenza*): ki jo lahko definiramo na štiri načine:
* *Krožnica s podanim središčem in polmerom* (*Dato il centro e il raggio*) – kar pomeni da jo lahko definiramo tako, da določimo polmer in središče krožnice.
* *Krožnica s podanim središčem čez točko* (*Dato il centro passante per un punto*) – dovolj je da podamo točko, ki je na krožnici in središče te.
* *Krožnica čez tri točke* (*Per tre punti*) – podamo tri točke na krožnici.
* *Krožnica z enačbo* (*Circonferenza con equazione*) – kjer določimo le *a*, *b* in *c*.
* Elipsa (*Elisse*): s predpisom – za katerega določimo *a* in *b*.
* Parabola (*Parabola*): s predpisoma in – za katera določimo *a*, *b* in *c*.
* Hiperbola (*Iperbola*): s predpisoma in – za katera določimo *a*, *b* in *c*.

V spodnji sliki (“Slika 19: Geometrjiski objekti”) je prikazano, kako se s programom nekateri geometrijski objekti izrišejo na zaslon.



Slika 19 : Geometrijski objekti

ZAKLJUČEK

Med spoznavanjem orodja sem doletela na nekaj dobrih in nekaj slabih plati programa. EffeDiX je res uporaben za analiziranje funkcij le, da mu po mojem mnenju manjka možnost računanja. V bistvu, če bi res želeli izračunati na primer nek pol, ničlo ali začetno vrednost neke funkcije, nam ne bi program tega omogočil. Nudi pa nam tabele, iz katerih lahko razberemo na zelo enostaven način vrednosti funkcije, ki nas zanimajo. Poleg tega ima zelo slabo dinamično geometrijo. Če primerjamo na primer program EffeDiX z GeoGebro, lahko opazimo, da v EffeDiX-u ni možno prosto risanje po zaslonu. Vse kar hočemo na zaslon narisati, moramo najprej definirati.

Kljub pomanjkljivostim je program zelo napreden. Če bi v programu omenjene pomanjkljivosti odpravili, bi bil res popoln in uporaben za vse.

VIRI

1. Lazzarini, P (2012), EffeDiX – software per la matematica, priročnik: <http://users.libero.it/prof.lazzarini/EffeDiX/Guida.pdf> (ogled 02.03.2015)
2. Spletna stran : <http://www.paololazzarini.it/EffeDiX/> (ogled 02.03.2015)