

Sprawozdanie z laboratorium:
Tutaj nazwa przedmiotu
(szablon)

Część I: Algorytmy optymalizacji lokalnej, problem QAP

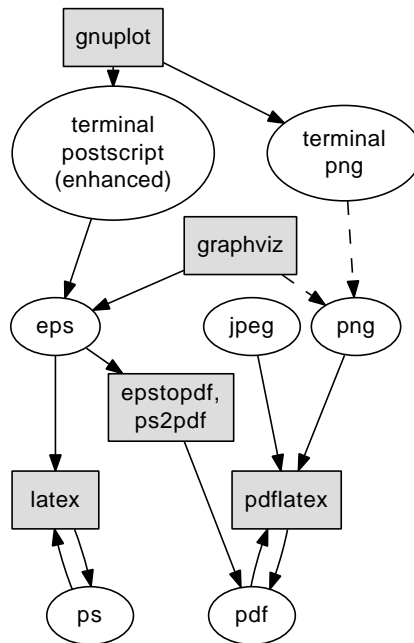
18 listopada 2017

Prowadzący: dr hab. inż. Maciej Komosiński

Autorzy: **Jan Kaczmarek** inf80123 ISWD jasiu@serwer.domena.poczta.pl
Ewa Kowalska inf89154 PIESI ewka@w.pl

Zajęcia poniedziałkowe, 11:00.

Oświadczam/y, że niniejsze sprawozdanie zostało przygotowane wyłącznie przez powyższych autora/ów, a wszystkie elementy pochodzące z innych źródeł zostały odpowiednio zaznaczone i są cytowane w bibliografii.



Rysunek 1: Przykładowy schemat z programu *graphviz* – narzędzia do automatycznego generowania schematów [1]. Przerywane strzałki oznaczają, że wszędzie gdzie się da używamy grafiki wektorowej – unikamy wstawiania bitmap do dokumentu. W niektórych przypadkach użycie bitmap jest uzasadnione (w celu szybkiego podglądu na ekranie lub dla niezwykle skomplikowanych grafik, zawierających np. setki tysięcy obiektów). Różnice w grafice rastrowej i wektorowej omawia prezentacja https://www.youtube.com/watch?v=_98SDNIpm24.

Udział autorów (jeśli > 1)

- JK zaimplementował..., przeprowadził eksperyment..., opisał..., przygotował...
- EK zaimplementowała..., przeprowadziła eksperyment..., opisała..., przygotowała...

1 Wstęp

To jest przykładowy tekst w LaTeX. Przeczytaj go uważnie (treść, jego źródło oraz %komentarze) i użyj tego źródła `*.tex` jako szablonu sprawozdania – to źródło pokazuje jak

- wstawić schemat stworzony *graphviz*’em (Rys. 1),
- wstawić wykres stworzony *gnuplot*’em (Rys. 2, 3, 4 i 5) oraz *matplotlib*’em (Rys. 6),
- zacytować literaturę sformatowaną przez *bibtex* [3, 2],
- odwoływać się do rysunków, cytowań i części sprawozdania (np. rozdział 2.1).

2 Cechy dobrego sprawozdania

Dobre sprawozdanie

- pozwala odtworzyć samodzielnie czytelnikowi eksperyment (od danych po wyniki),

- nie zawiera nieudomówień,
- przedstawia wnioski uporządkowane od ogólnych do szczegółowych,
- cytuje literaturę w tekście,
- nie zawiera zbyt obszernych listingów,
- czytelnie prezentuje wyniki – zwykle za pomocą wykresów,
- wszelkie dane liczbowe pokazuje z właściwą liczbą miejsc znaczących,
- jest zwięzłe i estetyczne.

2.1 Typografia

Pamiętajmy o różnicy pomiędzy łącznikiem¹ a myślnikiem – a także o cytowaniu wszelkich materiałów źródłowych w odpowiednich miejscach [5]. Cytujmy konkretną stronę, a nie ogólny adres witryny. Cudzysłowy polskie piszemy metodą „przecinków i apostrofów”.

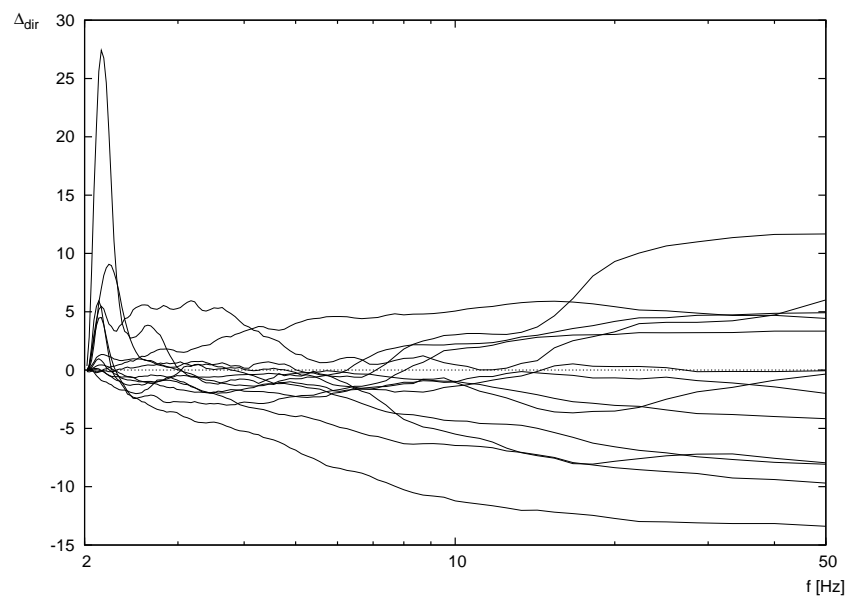
Do sprawdzania pisowni bezpośrednio w pliku .tex służy między innymi program *aspell*. Rozumie on różne sposoby kodowania polskich literek, a także ma wbudowane filtry do html’a i innych popularnych formatów. Dzięki tym filtrom pomija słowa kluczowe typowe dla danego formatu pliku, analizując tylko właściwy tekst.

3 Wykresy

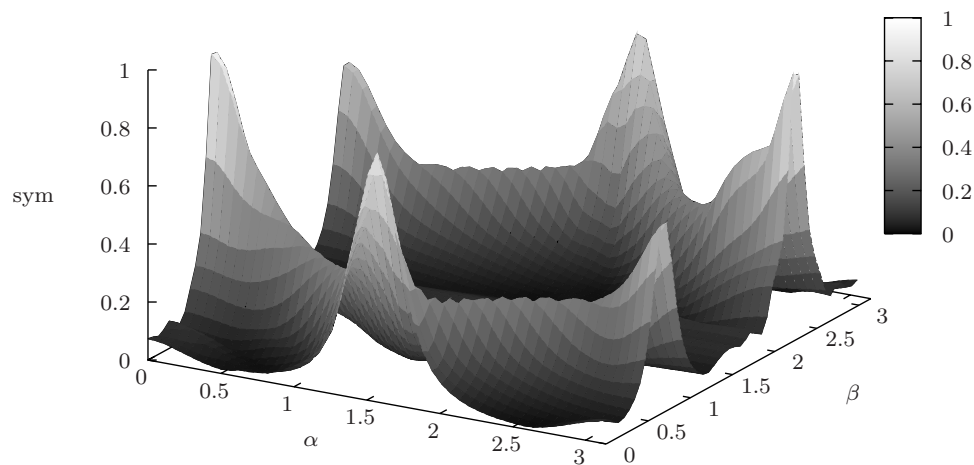
Do przetwarzania tekstowych plików z wynikami oraz rysowania wykresów wyśmienicie nadaje się python wzbogacony o bibliotekę matplotlib. Zdecydowanie warto się ich nauczyć! Jeśli jednak chciał(a)byś wykorzystać program gnuplot (co jest mniej przyszłościowe), to jego nowe wersje posiadają już niezły terminal ‘pdf’, więc można zrezygnować z pośrednictwa formatu ps/eps. Jeśli nie wiesz jak zrobić jakiś rodzaj wykresu, sprawdź stronę z przykładami gnuplota [4].

Zanim przygotujesz wykres, obejrzyj koniecznie porady dotyczące ich tworzenia – jak zrobić czytelny i profesjonalny wykres: <https://www.youtube.com/watch?v=pfSgcsQ2Mtk>.

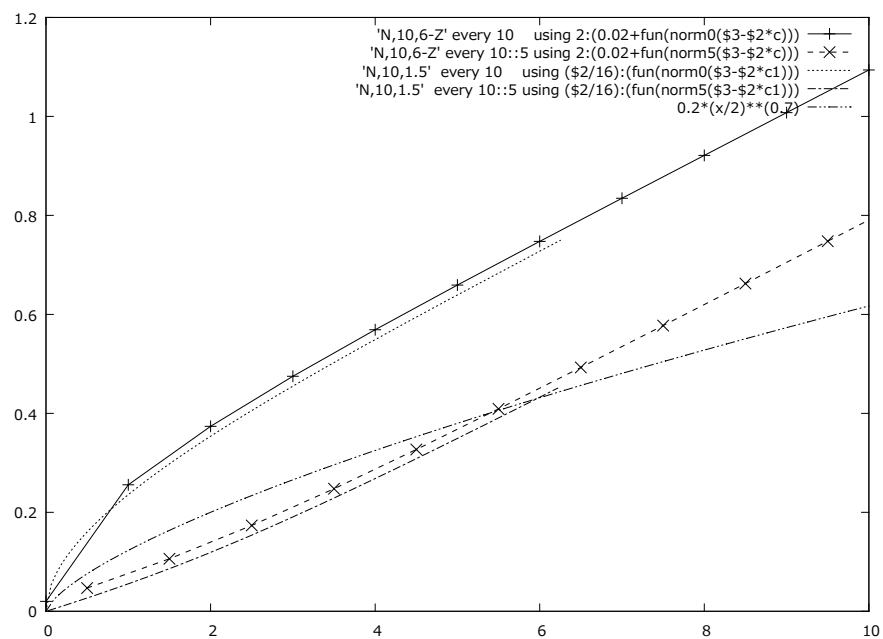
¹Przeczytaj w Wikipedii opis hasła „Dywiz”.



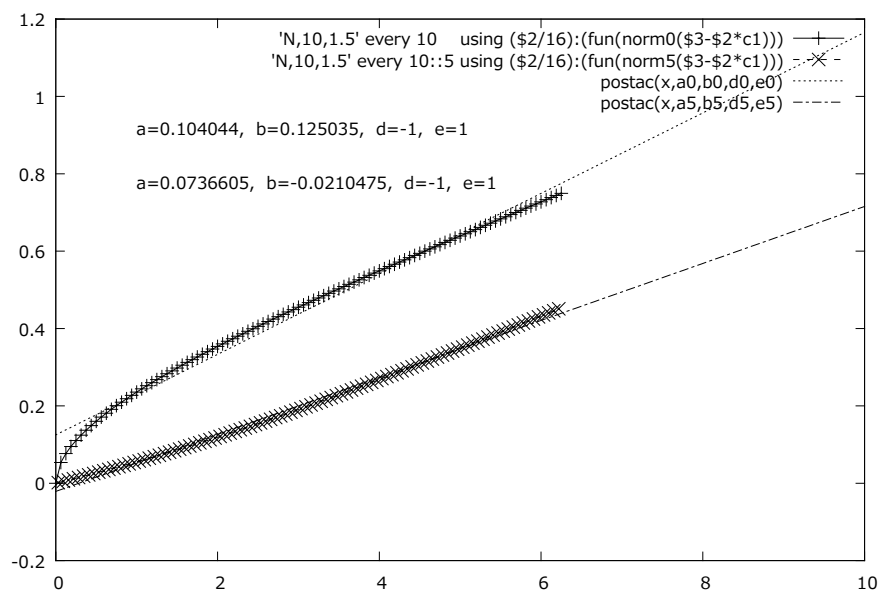
Rysunek 2: Przykładowy wykres z gnuplota, terminal postscript, zamieniony na pdf za pomocą programu eps2pdf z dystrybucji LaTeX'a (czasem eps2pdf). Wykres pokazuje różnice Δ_{dir} wartości p_{dir} dla kąta 90° .



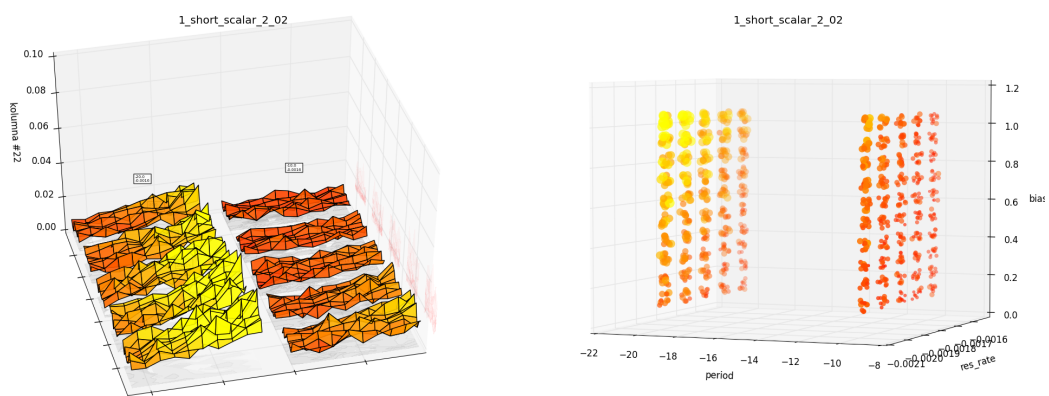
Rysunek 3: Jeszcze jeden przykładowy wykres z gnuplota.



Rysunek 4: Przykład filtrowania danych do wykresu (*every*) oraz użycie własnych funkcji i formuł w gnuplocie.



Rysunek 5: Przykład regresji: gnuplot ma wbudowany moduł dopasowujący do danych parametry funkcji o dowolnej zadanej postaci. Pozwala też definiować makra, prowadzić obliczenia i umieszczać na wykresie etykiety.



Rysunek 6: Przykład wizualizacji w python+matplotlib; tutaj dane 5D pokazane na dwa sposoby w 3D. Wstawione bitmapy (nieprawidłowo, powinna być postać wektorowa), a wykresy są tu za małe (nieczytelne).

Literatura

- [1] Emden R. Gansner and Stephen C. North. An open graph visualization system and its applications to software engineering. *Software – practice and experience*, 30(11):1203–1233, 2000. URL: <https://www.graphviz.org/>.
- [2] D. E. Goldberg. *The Design of Innovation: Lessons from and for Competent Genetic Algorithms*. Kluwer Academic Publishers, 2002.
- [3] Maciej Komosiński. Materiały do wykładu „Metaheurystyki i obliczenia inspirowane biologicznie”. Lecture notes, 2014.
- [4] Developers of gnuplot. gnuplot demo plots, 2011. <http://www.gnuplot.info/screenshots/index.html#demos>.
- [5] Wikipedia. Dash – wikipedia, the free encyclopedia, 2014. <http://en.wikipedia.org/wiki/Dash>.