**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

1. IDENTITAS PENGUSUL

Nama : **Munjiah Nur Sa’adah**

NRP : **5108 100 167**

Dosen Wali : **Ir. F. X. Arunanto, M.Sc**

1. JUDUL TUGAS AKHIR

***“Implementasi Fungsi Pembangkit Numerik dengan Metode Segmentasi Piecewise Polynomial”***

1. LATAR BELAKANG

*Function generator* adalah bagian dari peralatan elektronik atau perangkat lunak yang digunakan untuk menghasilkan berbagai jenis bentuk gelombang listrik melalui berbagai frekuensi. Salah satu kelebihan dari *function generator* adalah kemampuannya untuk mengunci fase gelombang ke sumber sinyal eksternal atau *function generator* lain. Beberapa bentuk gelombang yang paling umum dihasilkan oleh *function generator* adalah sinus, persegi, segitiga, dan bentuk gelombang gigi gergaji.

*Piecewise function* adalah fungsi yang definisinya tergantung pada variabel independennya. Polinomial sendiri adalah ekspresi terbatas yang dibangun dari beberapa variabel dan konstanta menggunakan hanya operasi penjumlahan, pengurangan, perkalian, dan pangkat non-negatif bilangan bulat. Jadi, *piecewise polynomial* adalah fungsi polinomial yang definisinya tergantung pada variabel independen.

Beberapa algoritma segmentasi fungsi polinomial yang telah ada sebelum algoritma ini adalah algoritma segmentasi non-uniform dan algoritma segmentasi uniform. Segmentasi non-uniform adalah segmentasi dengan memilih segmen yang memiliki lebar segmen paling luas sehingga pendekatan tersebut tidak melebihi pendekatan error yang jauh dari nilai yang tepat. Pada segmentasi non-uniform diasumsikan lebar tiap segmen berbeda-beda. Sedangkan segmentasi uniform biasanya digunakan pada segmen yang memiliki lebar sama. Sebagai contoh, jika lebar segmen adalah 2*m*, maka encoder indeks segmen dapat dihapus dan urutan bit yang lebih tinggi, *n-m*, digunakan untuk menggerakkan alamat encoder indeks segmen, dimana *n* adalah jumlah bit untuk mengkodekan *x*. Namun, terdapat kekurangan pada algoritma-algoritma tersebut, seperti pada algoritma segmentasi non-uniform menghasilkan segmentasi yang tidak optimum[2, 3] dan pada segmentasi uniform dan non-uniform menghasilkan waktu komputasi yang memakan waktu selama beberapa hari atau beberapa minggu[4].

1. RUMUSAN MASALAH

Rumusan masalah yang diangkat dalam Tugas Akhir ini adalah sebagai berikut :

* + 1. Memahami konsep segmentasi pada fungsi polinomial.
    2. Mendesain algoritma pada MATLAB untuk diketahui kinerjanya.

1. BATASAN MASALAH

Permasalahan yang dibahas pada Tugas Akhir ini memiliki beberapa batasan, yaitu sebagai berikut :

* + 1. Simulasi eksperimen dilakukan menggunakan MATLAB 7.6.0 atau MATLAB R2008a.
    2. Perkiraan polinomial hanya fokus pada perkiraan linear.

1. TUJUAN TUGAS AKHIR

Tugas akhir pada Tugas Akhir ini adalah, sebagai berikut :

1. Mengimplementasi algoritma yang dapat melakukan segmentasi pada fungsi polinomial.
2. Mengetahui kinerja algoritma dalam melakukan segmentasi pada fungsi polinomial.
3. Membandingkan kinerja algoritma ini dengan metode *brute force*.
4. MANFAAT TUGAS AKHIR

Tugas akhir ini dikerjakan dengan harapan dapat memberikan manfaat yang besar pada bidang informatika dalam mensegmentasi fungsi polinomial sehingga akan mempermudah dalam pengolahan data berikutnya.

1. RINGKASAN TUGAS AKHIR

Dalam algoritma segmentasi potongan fungsi polinomial terdapat langkah-langkah sebagai berikut :



1. ESTIMATE

Pada bagian pertama ini, lebar segmen diestimasi. Berdasarkan lebar segmen ini, *end point* dapat ditemukan dan perkiraan error untuk segmen yang diusulkan dapat dihitung.

1. LOCATE

Pada bagian kedua ini, dua titik pada domain yang terletak sedemikian rupa sehingga satu titik menghasilkan segmen yang perkiraan error berada di bawah (atau sama dengan) ε dan titik lainnya menghasilkan sebuah segmen yang perkiraan error berada di atas ε.

Pengertian di atas dapat disederhanakan seperti ini, berdasarkan estimasi lebar segmen yang telah dihitung dalam ESTIMATE, diketahui apakah titik yang sesuai berada di atas lebar segmen optimal atau berada di bawahnya (atau sama dengan). Jika berada di atas, LOCATE menghasilkan nilai-nilai yang lebih rendah dari *x* untuk mencari dua titik yang berjarak jauh dari lebar segmen optimal. Jika berada di bawah, LOCATE menghasilkan nilai-nilai yang lebih tinggi. Asumsi bahwa titik berada di bawah lebar segmen optimal. Maka algoritma menghasilkan lebar segmen optimum dengan satu titik awal. Kemudian menghitung perkiraan error untuk segmen baru, dan jumlah langkah telah bertambah 1. Jika perkiraan error melebihi ε, berhentilah proses ESTIMATE dan tidak perlu melangkah ke proses berikutnya, yaitu PINPOINT.

Namun, jika perkiraan error kurang dari (atau sama dengan) ε, maka algoritma akan maju 2 poin. Mengulangi menghitung perkiraan error dan jumlah langkah bertambah 1. Proses ini akan diulang secara terus menerus, kecuali bila algoritma maju sebesar 4, 8, dll poin hingga ditemukan 2 titik yang berada pada masing-masing sisi lebar segmen optimal. Pada akhir proses ESTIMATE, dua titik terakhir dianggap sebagai H dan L yang sesuai dengan titik akhir segmen yang mempunyai jarak cukup jauh dari *end point* yang tepat. Secara terperinci, H adalah *end point* segmen yang mempunyai perkiraan error lebih besar dari ε dan L adalah *end point* segmen yang mempunyai perkiraan error kurang dari atau sama dengan ε.

1. PINPOINT

Pada bagian ketiga ini, H dan L dibagi seperti ini, , dan didapatkan titik A sebagai titik tengah H dan L. Kemudian dibuatlah sebuah segmen baru dengan *end point* A dan menghitung perkiraan errornya. Jika perkiraan error tersebut melebihi ε, maka H digantikan dengan A dan proses akan berulang kembali. Jika sebaliknya, perkiraan error tersebut kurang dari atau sama dengan ε, maka L digantikan dengan A dan proses akan berulang kembali. Setiap kali menghitung perkiraan error, maka jumlah langkah bertambah 1. Proses ini akan berhenti bila ditemukan H dan L yang berdekatan. L dipilih menjadi *end point* segmen karena maksimum error pada segmen berakhir di L kurang dari atau sama dengan ε, sedangkan maksimum error pada segmen berakhir di H jika melebihi ε.

1. METODOLOGI

Metodologi yang akan dilakukan dalam Tugas Akhir ini memiliki beberapa tahapan, diantaranya sebagai berikut :

1. Penyusunan Proposal Tugas Akhir

Tahap awal untuk memulai pengerjaan Tugas Akhir adalah penyusunan Proposal Tugas Akhir. Pada proposal ini, penulis mengajukan gagasan implementasi fungsi pembangkit numerik dengan metode segmentasi *piecewise polynomial*.

1. Studi Literatur

Pada tahap ini dilakukan pencarian informasi dan studi literatur yang diperlukan untuk pengumpulan data dan desain sistem yang akan dibuat. Informasi didapatkan dari buku dan materi-materi lain yang berhubungan dengan algoritma metode yang digunakan dalam pengerjaan tugas akhir ini, yang didapat dari *internet* maupun buku acuan.

1. Implementasi

Implementasi merupakan tahap untuk membangun algoritma tersebut. Untuk membangun algoritma yang telah dirancang sebelumnya, diimplementasikan dengan menggunakan MATLAB.

1. Pengujian dan Evaluasi

Pada tahap ini dilakukan uji coba dengan menggunakan 15 fungsi numerik untuk mencoba jalannya aplikasi apakah telah sesuai dengan rancangan dan desain implementasi yang dibuat, juga untuk mencari kesalahan – kesalahan program yang mungkin terjadi untuk selanjutnya dapat dilakukan penyempurnaan.

1. Penyusunan Buku Tugas Akhir

Tahap terakhir merupakan penyusunan laporan yang memuat dokumentasi mengenai pembuatan serta hasil dari implementasi perangkat lunak yang telah dibuat.

1. JADWAL KEGIATAN TUGAS AKHIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Tahapan** | **Bulan** | | | | | | | | | | | | | | | |
| **September** | | | | **Oktober** | | | | **Nopember** | | | | **Desember** | | | |
| 1. | Penyusunan Proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2. | Studi Literatur |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3. | Implementasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. | Pengujian dan Evaluasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5. | Penyusunan Buku Tugas Akhir |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. DAFTAR PUSTAKA
   1. A.G. Bromley, The evolution of Babbage’s calculating engines, Ann. Hist. Comput. 9 (1987) 113–136.
   2. D.U. Lee, Wayne Luk, J. Villasenor, P.Y.K. Cheung, Non-uniform segmentation for hardware function evaluation, in: Proc. Inter. Conf. on Field Programmable Logic and Applications, Lisbon, Portugal, September, 2003, pp. 796–807.
   3. T. Sasao, J.T. Butler, M.D. Riedel, Application of LUT cascades to numerical function generators, in: The 12th Workshop on Synthesis and System Integration of Mixed Information technologies, SASIMI2001, Kanazawa, Japan, October 18-19, 2004, pp. 442-429.
   4. C.L. Frenzen, T. Sasao, J.T. Butler, On the number of segments needed in a piecewise linear approximation, J. Comput. Appl.Math. 234 (2010) 437–466.

**LEMBAR PENGESAHAN**

**Surabaya, 13 Oktober2011**

Menyetujui,

Dosen Pembimbing 1 Dosen Pembimbing 2

**(Yudhi Purwananto, S.Kom, M.Kom.) (Rully Soelaiman, S.Kom, M.Kom)**

**(NIP. 197007141997031002) (NIP. 197002131994021001)**