

Penerapan Pewarnaan Simpul Graf dengan Algoritma *Welch Powell* dan Algoritma *Depth First Search* pada Distribusi Giliran Main Angklung

Fithratulhay Pribadi^{1*}, Rinaldi Munir¹

¹Program Studi Teknik Informatika, Institut Teknologi Bandung
pribadi191200@gmail.com (*corresponding author), rinaldi@informatika.org

Abstrak. Angklung adalah alat musik yang setiap alat musiknya hanya bisa memainkan satu nada saja. Untuk memainkan sebuah lagu dengan angklung, membutuhkan sebuah tim yang masing-masing pemainnya memegang satu atau lebih angklung. Pengaturan giliran memainkan dalam sebuah lagu agar pemain mengetahui ketukan dan nada apa yang harus dimainkannya, dapat disebut pembagian giliran angklung. Pada saat ini, masih sering terjadi gap jumlah turn antar pemain. Untuk mengatasi masalah tersebut, telah dibuat sebuah aplikasi untuk melakukan proses pembagian giliran angklung yang memanfaatkan pewarnaan graf dengan algoritma *Welch Powell* untuk membagi giliran menjadi kelompok-kelompok dimana tidak ada giliran lain yang dimainkan secara bersamaan (*clashing*) dan dilanjutkan dengan Algoritma *Depth First Search* yang akan mengoptimalkan jumlah ketukan di setiap grup. Aplikasi tersebut mampu membuat proses distribusi menjadi lebih efisien dan cukup memuaskan. Namun perbaikan masih diperlukan karena ada beberapa aspek yang belum dijadikan parameter dalam aplikasi.

Kata kunci: Angklung, *Depth First Search*, Pewarnaan Graf.

Abstract. Angklung is a musical instrument in which each instrument can only play one note. To play a song with angklung requires a team in which each player holds one or more angklung. Setting the turn to play in a song so that the player knows on what beat and what note he should play can be called the distribution of angklung's turn. At this time, there is still often a gap in the number of turns between players. To overcome this problem, an application has been created to perform the angklung turn distribution process that utilizes graph coloring with the *Welch Powell* algorithm to divide the turn into groups where there are no other turns to play at the same time (*clashing*) and continued with *Depth First Search* algorithm that will optimize the number of beats in each group. The application is able to make the distribution process more efficient and quite satisfying. However, improvements are still needed because there are several aspects that have not been used as parameters in the application.

Keywords: Angklung, *Depth First Search*, Graph Coloring.

Received August 2021 / **Revised** October 2021 / **Accepted** December 2021

This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).



PENDAHULUAN

Angklung merupakan salah satu alat musik tradisional Indonesia yang berasal dari Tanah Sunda [1]. Ilustrasi sebuah angklung dapat dilihat pada Gambar 1. Berbeda dengan alat musik pada umumnya yang dapat membunyikan seluruh nada pada satu alat, satu angklung hanya dapat membunyikan satu nada. Oleh karena itu, untuk memainkan sebuah lagu utuh dengan angklung dibutuhkan sebuah tim dengan jumlah pemain sekitar 10 hingga 40 orang yang memegang satu atau lebih angklung. Setiap pemain akan bermain sesuai gilirannya masing-masing. Pengaturan giliran main pada sebuah lagu sehingga pemain tahu pada ketukan berapa dan nada apa ia harus bermain dapat disebut sebagai distribusi giliran main angklung.



Gambar 1. Ilustrasi angklung [2]

Distribusi giliran main angklung yang ada saat ini sudah diusahakan untuk mencapai kondisi ideal. Namun, pada praktiknya di lapangan, masih sering terdapat kesenjangan jumlah giliran main antarpemain karena masih ada tahap yang bergantung pada intuisi orang yang melakukannya. Oleh karena itu, perlu dicari solusi agar hasil dari distribusi giliran main angklung lebih mendekati kondisi ideal di mana setiap pemain pada suatu tim memiliki jumlah giliran main yang mendekati rata-rata jumlah giliran main pada tim tersebut. Sebelumnya, telah terdapat penelitian-penelitian yang berkaitan dengan distribusi giliran main angklung dan mempunyai kelebihan masing-masing. Namun, hasil dari penelitian-penelitian tersebut masih sangat memungkinkan kesenjangan giliran main antarpemain terjadi. Untuk itu, diperlukan modifikasi dari metode yang sudah ada sebelumnya untuk mengatasi masalah tersebut.

A. Partitur Angklung

Partitur angklung pada umumnya ditulis menggunakan nada relatif. Nada relatif adalah nada yang frekuensinya bisa berubah-ubah tergantung pada nada dasarnya. Nada relatif biasa ditulis dalam bentuk angka yang dibaca sebagai do, re, mi, fa, sol, la, dan si. Kenaikan setengah nada akan diberi tanda “/”, sedangkan penurunan setengah nada akan diberi tanda “\”. Setiap berganti ke oktaf yang lebih tinggi, akan ditambahkan satu titik di atas nada tersebut, begitu pula saat berganti ke oktaf yang lebih rendah, akan ditambahkan satu titik di bawahnya. Mengambil referensi dari Keluarga Paduan Angklung Institut Teknologi Bandung (KPA-ITB), partitur ditulis menggunakan *Microsoft Excel*. Satu sel pada *Microsoft Excel* bermakna satu ketukan. Jika hanya terdapat satu nada pada satu sel, berarti nada tersebut dimainkan sebanyak 1 ketukan. Jika setelah sel tersebut diisi dengan tanda titik (“.”), berarti nada tersebut dipanjangkan sebanyak ketukan pada titik tersebut. Satu garis yang tertulis di atas nada menandakan nada tersebut dimainkan sebanyak setengah ketuk, sedangkan dua garis menandakan nada dimainkan sebanyak seperempat ketuk.

B. Distribusi Giliran Main Angklung

Distribusi giliran main angklung merupakan pengaturan giliran main pada sebuah lagu sehingga pemain tahu pada ketukan berapa dan nada apa ia harus bermain. Pengaturan giliran main ini harus memenuhi kriteria tertentu untuk mencapai hasil yang ideal. Berdasarkan pengamatan yang telah dilakukan penulis pada tim angklung KPA-ITB, kriteria ideal dalam melakukan distribusi giliran main angklung adalah membuat setiap pemain pada suatu tim memiliki jumlah giliran main yang mendekati rata-rata jumlah giliran main pada tim tersebut dan meminimalkan jumlah angklung yang dipegang pemain. Pemain yang mendapat giliran main terlalu sedikit dapat merasa tidak senang karena banyak menganggur, sedangkan pemain yang terlalu banyak bermain mendapat beban berlebih yang seharusnya dapat diberikan kepada pemain yang terlalu sedikit bermain. Memberi banyak angklung kepada seorang pemain tidak serta merta membuat jumlah giliran main menjadi lebih banyak, karena bisa saja terjadi bentrok.

Bentrok adalah kondisi di mana terdapat lebih dari satu angklung yang harus dimainkan bersamaan atau berdekatan sehingga ada angklung yang harus dikorbankan dalam arti tidak dapat dimainkan [3]. Pada partitur, bentrok dapat terlihat jika pasangan angklung tertulis pada kolom yang sama (dimainkan pada ketukan yang sama) atau saling bersebelahan (dimainkan sebelum atau sesudahnya). Sebagai contoh, dapat dilihat pada Gambar 2 bahwa angklung pada cell yang diberi warna abu akan bentrok dengan angklung lain pada cell yang diberi warna merah. Selain itu, jumlah angklung yang dipegang setiap pemain juga perlu diminimalkan untuk meringankan beban yang dibawa pemain dalam penampilan.

55	0		4	3	25
55	0		4	3	25
.	2	.	6	5	2
.	7	.	1	2	7
.	5	.	1	5	5

Gambar 2. Contoh bentrok

C. Algoritma Welch Powell

Terdapat tiga macam persoalan pewarnaan graf, salah satunya adalah pewarnaan simpul graf. Pewarnaan simpul graf merupakan pemberian warna pada simpul graf, sedemikian rupa sehingga dua simpul yang

bertetangga memiliki warna yang berbeda [4]. Jumlah minimal dari banyaknya warna yang dibutuhkan untuk mewarnai graf G disebut dengan bilangan kromatik dan dapat dilambangkan dengan $\chi(G)$ [5]. Tujuan dari pewarnaan simpul graf merupakan pencarian bilangan kromatik tersebut.

Salah satu algoritma yang dapat digunakan untuk melakukan pewarnaan simpul graf merupakan Algoritma *Welch Powell*. Untuk dapat menggunakan algoritma ini, perlu diketahui terlebih dahulu matriks *adjacency* dan derajat simpul. Matriks *adjacency* adalah matriks yang merepresentasikan status ketetanggaan antar simpul dalam sebuah graf. Pada sebuah matriks *adjacency* M , nilai tiap $M[i, j]$ merupakan hasil persamaan (1).

$$M[i, j] = \begin{cases} 1, & \text{terdapat sisi di antara} \\ & \text{simpul } i \text{ dan } j \\ 0, & \text{sebaliknya} \end{cases} \quad (1)$$

Setelah diketahui matriks *adjacency* suatu graf beserta derajat tiap simpulnya, Algoritma *Welch Powell* dapat dilakukan dengan tahapan sebagai berikut [7]:

1. Urutkan simpul dari derajat terbesar menuju derajat terkecil.
2. Beri warna pada simpul dengan derajat terbesar yang belum memiliki warna.
3. Cari simpul yang belum diberi warna dari matriks *adjacency* di mana simpul tersebut tidak bertetangga dengan simpul yang diberi warna pada langkah nomor 2 dan masukkan simpul-simpul tersebut ke dalam sebuah himpunan V' .
4. Beri warna pada simpul dengan derajat tertinggi dalam himpunan V' dengan warna yang sama pada langkah 2 dan hapus simpul-simpul yang bertetangga dengan simpul tersebut dari himpunan V' . Ulangi langkah ini hingga simpul yang ada dalam himpunan V' sudah diberi warna semua.

Jika terdapat simpul yang masih belum memiliki warna, ulangi langkah 2 hingga 4 dengan warna yang baru. Jika semua simpul sudah memiliki warna, maka proses pewarnaan simpul graf telah selesai.

D. Algoritma *Depth First Search*

Algoritma *Depth First Search*, yang biasa disingkat menjadi DFS, merupakan algoritma pencarian pencarian solusi pada sebuah pohon yang menelusuri satu cabang sebuah pohon sampai menemukan solusi yang diinginkan [6]. Algoritma ini akan menelusuri simpul paling kiri dari setiap level. Jika simpul paling kiri pada level terdalam bukan merupakan solusi, maka pencarian dilanjutkan ke simpul yang ada di kanannya. Jika solusi masih juga belum ditemukan, maka pencarian akan kembali menuju level sebelumnya. Proses tersebut akan terus diulang hingga menemukan solusi.

Terdapat dua kelebihan algoritma DFS [7]:

1. Memori yang dibutuhkan lebih kecil karena hanya menyimpan jalur menuju simpul yang masih aktif.
2. Pada pohon dengan banyak percabangan, algoritma DFS memungkinkan ditemukannya solusi tanpa menguji banyak jangkauan pencarian karena tidak perlu menguji seluruh simpul pada level tertentu.

E. Penelitian Model Matematika dalam Menentukan Distribusi Angklung pada Suatu Penampilan

Distribusi giliran main angklung merupakan pengaturan giliran main pada sebuah lagu sehingga pemain tahu pada ketukan keberapa dan nada apa ia harus bermain. Pengaturan giliran main ini harus memenuhi kriteria tertentu

Penelitian Model Matematika Dalam Menentukan Distribusi Angklung Pada Suatu Penampilan dilakukan oleh Riska Aditya Puspa Kania pada tahun 2011 [8]. Penelitian ini berusaha melakukan automasi proses tonjur (menghitung bermainnya angklung dalam suatu lagu [9]) dan distribusi angklung dengan permodelan graf. Pada model graf yang digunakan, simpul mewakili nomor angklung dan sisi mewakili kebentrokan. Proses tonjur dilakukan dengan membuat matriks ketetanggaan dari model graf yang dibentuk di awal. Setelah didapatkan matriks ketetanggaan, dilakukan proses distribusi angklung dengan memanfaatkan proses pewarnaan simpul graf. Minimum warna yang digunakan merepresentasikan jumlah pemain minimum yang dibutuhkan.

Pada penelitian ini, automasi berhasil dilakukan. Namun, karena proses-proses tersebut hanya mempertimbangkan kebentrokkan pasangan angklung sepanjang lagu, kesenjangan jumlah giliran main antarpemain masih mungkin terjadi.

METODE

A. Analisis Solusi

Untuk menyelesaikan permasalahan kesenjangan pada distribusi giliran main angklung, dapat dilakukan modifikasi dari penelitian terkait dengan membuat sistem yang mengelompokkan bukan berdasarkan kebentrokkan nomor angklung, melainkan kebentrokkan giliran main. Metode pengelompokkan yang direncanakan pada solusi ini tetap menggunakan pewarnaan graf. Algoritma yang akan digunakan pada pewarnaan graf tersebut adalah algoritma *Welch Powell*. Dengan mengelompokkan berdasarkan kebentrokkan giliran main, nomor angklung yang bermain penuh selama satu lagu nantinya dapat dibagi giliran mainnya kepada beberapa pemain, sehingga jumlah giliran main akan menjadi lebih seimbang.

Setelah dikelompokkan, diperlukan adanya penyeimbangan jumlah giliran main antarpemain dan proses yang meminimalkan jumlah nada (angklung) yang dipegang oleh tiap pemain. Untuk proses ini, algoritma yang direncanakan untuk digunakan adalah algoritma *Depth First Search*. Dengan begitu, akan dihasilkan kelompok giliran main dengan jumlah giliran main antarkelompok yang seimbang dan jumlah angklung yang minimal pada tiap kelompok. Kelompok giliran main inilah yang akan dibagikan kepada setiap pemain.

B. Deskripsi Solusi

Proses penyusunan distribusi giliran main angklung akan dilakukan mengikuti langkah-langkah sebagai berikut:

1. Menerima masukan partitur lagu, jumlah maksimal pemain angklung, dan jumlah maksimal angklung yang dipegang masing-masing pemain.
Partitur lagu dibutuhkan karena hasil distribusi giliran main angklung akan berbeda pada setiap lagu. Jumlah maksimal pemain dibutuhkan agar distribusi yang dihasilkan tidak melebihi kapasitas bermain suatu tim, sedangkan jumlah maksimal angklung yang dipegang pemain dibutuhkan untuk menghindari kejadian pemain memegang terlalu banyak angklung.
2. Memindai partitur untuk mengetahui data giliran main dan kebentrokkan antar giliran main yang ada.
Pada tahap ini, dilakukan pencatatan seluruh giliran main yang ada pada partitur yang menjadi masukan. Bentuk dari giliran main adalah objek yang berisi informasi nada yang dimainkan, pada ketukan berapa ia mulai bermain, dan berapa ketukan nada tersebut dimainkan.
Setelah terdapat data giliran main, dilakukan pencatatan kebentrokkan antar giliran main. Setiap giliran main akan dianggap bentrok dengan giliran main yang dimainkan pada ketukan yang sama, pada ketukan sebelumnya, atau pada ketukan setelahnya.
3. Mengelompokkan giliran main yang tidak bentrok menggunakan pewarnaan graf dengan algoritma *Welch Powell*.
Setelah terdapat data giliran main, dilakukan pengelompokkan giliran main yang tidak bentrok menggunakan pewarnaan graf dengan algoritma *Welch Powell*. Detail dari tahap ini adalah sebagai berikut:
 - a. Merepresentasikan data giliran main beserta kebentrokannya ke dalam bentuk graf, dengan simpul mewakili data giliran main dan sisi mewakili kebentrokkan antar giliran main.
 - b. Mengurutkan simpul dari derajat terbesar menuju derajat terkecil.
 - c. Memberi warna baru pada simpul dengan derajat terbesar yang belum memiliki warna.
 - d. Mencari simpul yang belum diberi warna dari matriks *adjacency* di mana simpul tersebut tidak bertetangga dengan simpul yang diberi warna pada langkah poin c dan masukkan simpul-simpul tersebut ke dalam sebuah himpunan V' .
 - e. Memberi warna pada simpul dengan derajat tertinggi dalam himpunan V' dengan warna yang sama pada langkah poin c dan hapus simpul-simpul yang bertetangga dengan simpul tersebut dari himpunan V' . Jika kelompok warna sudah memiliki jumlah maksimal nada (angklung) yang telah ditentukan, maka simpul dengan nada yang belum terdapat pada kelompok warna juga dihapus dari himpunan V' .
 - f. Mengulangi langkah pada poin e hingga simpul yang ada dalam himpunan V' sudah diberi warna semua.

Jika masih terdapat simpul yang belum memiliki warna, maka langkah poin c hingga poin f terus diulang. Jika semua simpul sudah memiliki warna, maka tahap pengelompokan giliran main yang tidak bentrok selesai. Dengan prinsip pewarnaan graf, pasangan giliran main yang diberi warna yang sama adalah pasangan giliran main yang tidak bentrok sepanjang lagu.

4. Penyeimbangan jumlah giliran main dan meminimalan jumlah nada (angklung) pada setiap kelompok dengan algoritma *Depth First Search*.

Setelah dihasilkan kelompok-kelompok giliran main yang tidak bentrok, dilakukan penyeimbangan jumlah giliran main dan meminimalan jumlah nada pada setiap kelompok dengan penelusuran pohon pencarian solusi yang dibentuk secara dinamis dengan algoritma *Depth First Search*. Pada pohon pencarian tersebut, dilakukan pemindahan giliran main dengan tetap memastikan tidak ada bentrok yang terjadi. Detail dari tahap ini adalah sebagai berikut:

- a. Mendata jumlah ketukan serta nada yang ada di setiap kelompok giliran main dan menghitung rata-rata jumlah ketukan.
- b. Menambah giliran main opsional pada tiap kelompok giliran main. Giliran main opsional adalah giliran main yang tidak wajib dimainkan, namun dapat dimainkan karena memiliki nada (angklung) yang sama dengan giliran main utama yang dibagikan pada pewarnaan graf. Penambahan giliran main opsional dilakukan dengan ketentuan sebagai berikut:
 - 1) Total ketukan pada kelompok tersebut belum melebihi rata-rata yang didapat pada poin a.
 - 2) Giliran main tidak bentrok dengan giliran main yang sudah ada pada kelompok tersebut.
 - 3) Nada (angklung) sudah terdapat pada kelompok tersebut.
- c. Jika standar deviasi jumlah ketukan tiap kelompok lebih kecil dari satu dan jumlah nada (angklung) yang ada pada setiap kelompok giliran main tidak melebihi jumlah maksimal, maka proses sudah selesai. Jika syarat tersebut belum terpenuhi, maka giliran main opsional yang ditambahkan pada langkah b dihapus. Setelah itu, proses dilanjutkan menuju langkah d.
- d. Memilih kelompok giliran main dengan jumlah ketukan terbanyak untuk diproses pada langkah selanjutnya. Jika terdapat lebih dari satu kelompok giliran main dengan jumlah ketukan terbanyak, maka yang dipilih di antaranya adalah kelompok giliran main dengan jumlah nada terbanyak.
- e. Pada kelompok yang sudah dipilih pada langkah d, pilih giliran main dengan jumlah ketukan terbanyak yang tidak bentrok dengan giliran main yang ada pada kelompok dengan jumlah ketukan paling sedikit dan nada pada giliran main sudah ada pada kelompok dengan jumlah ketukan paling sedikit. Setelahnya, pindahkan giliran main tersebut ke kelompok dengan jumlah ketukan paling sedikit.
- f. Jika tidak ada giliran main yang memenuhi ketentuan pada langkah d, maka kelompok giliran main dengan jumlah ketukan paling sedikit diganti dengan kelompok giliran main dengan jumlah nada paling sedikit setelahnya lalu ulangi langkah d. Jika benar-benar tidak ada yang memenuhi, maka buat kelompok baru dan pindahkan giliran main tersebut ke kelompok baru.
- g. Mengulangi proses dengan kembali ke langkah a.
Jika jumlah kelompok giliran main melebihi batas yang ditentukan (tergantung pada jumlah maksimal pemain), maka dilakukan *backtracking* dengan kembali pada proses putaran sebelumnya dan memilih kelompok giliran main yang lain pada langkah c. Jika seluruh kemungkinan telah dilakukan dan proses belum mencapai tujuan yang diinginkan (standar deviasi jumlah ketukan dan nada (angklung) yang ada pada setiap kelompok giliran main tidak melebihi tiga) maka hasil akhir adalah kondisi terbaik dari kondisi-kondisi yang ada saat *backtracking* dilakukan.

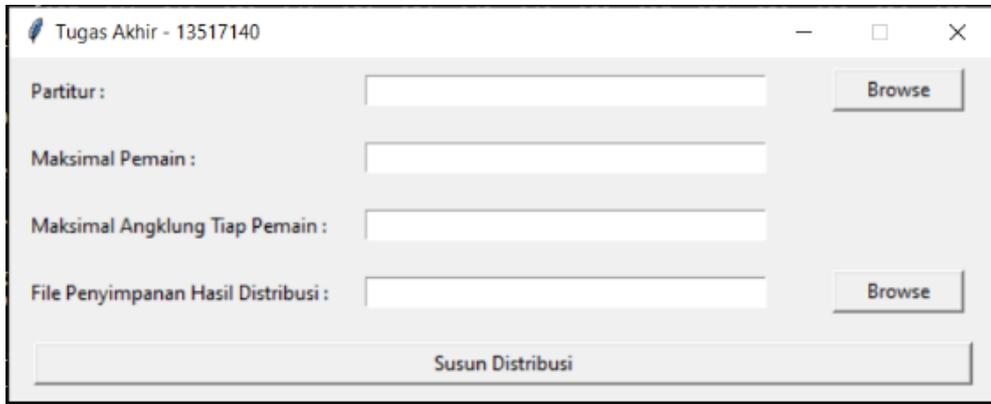
5. Pembuatan partitur angklung baru untuk setiap kelompok giliran main.

Agar mudah dipahami pemain, distribusi giliran main angklung yang dihasilkan dari tahap sebelumnya diubah terlebih dahulu ke dalam bentuk partitur yang sudah diberi tanda di mana saja pemain harus memainkan angklungnya. Satu kelompok giliran main akan membentuk sebuah partitur. Pada tahap ini, pertama-tama dilakukan duplikasi partitur angklung yang menjadi masukan di tahap awal. Setelah itu, sistem akan memberi tanda pada giliran main yang ada pada kelompok giliran main yang sudah ditetapkan pada partitur baru tersebut.

HASIL DAN PEMBAHASAN

A. Hasil

Pengembangan aplikasi dilakukan menggunakan bahasa pemrograman Python pada sistem operasi Microsoft Windows. Antarmuka utama aplikasi ini dapat dilihat pada Gambar 3, sedangkan contoh hasil distribusi dapat dilihat pada Gambar 4.



Gambar 3. Antarmuka utama

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	
1																													
2				Angklung : 13																									
3				Total Ketukan : 16.0																									
4				Do = G(no.1), 4/4																									
5			5	3.4	5	1.2	3	.	i	.	i	.7	2i	76	5	.	.	0	5	3.4	5	1.2							
6			3	1.2	3	3	5	.	3	.	4	.	4	4	3	.	.	0	3	1.2	3	3							
7																													
8			3	.	i	.	2	.2	3	4	5	.	.	0	2	2.2	3	3	4	.3	4	.							
9			3	.	6	.	7	.	i	i	7	.	.	0	7	7.7	i	i	2	.3	2	.							
10							2	.	2	.	5	.	.	0	5	.	5	5	5	5	.	05	67						
11																													
12			3	3.2	i	3	2	.	.	0	5	1.2	3	5	5	.	4	3.2	i	.	2	.							
13			5	5.5	5	5	5	.	.	0	3	5.5	5	7	7	.	6	3	5	5	.	5	.						
14			1	1	1	1	7	.	.	0										3	.	7	.						
15																													
16			3	.	.	0	5	1.2	3	5	5	.	4	3.2	3	.	2	.	i	.	.	.	0						
17			5	.	.	0	5	5.5	5	7	7	.	6	3	5	.	4	.	3	.	.	.	0						
18			1	7	6	5	3	.	5	3	3	.	2	2	1	.	7	.	1	.	.	.	0						

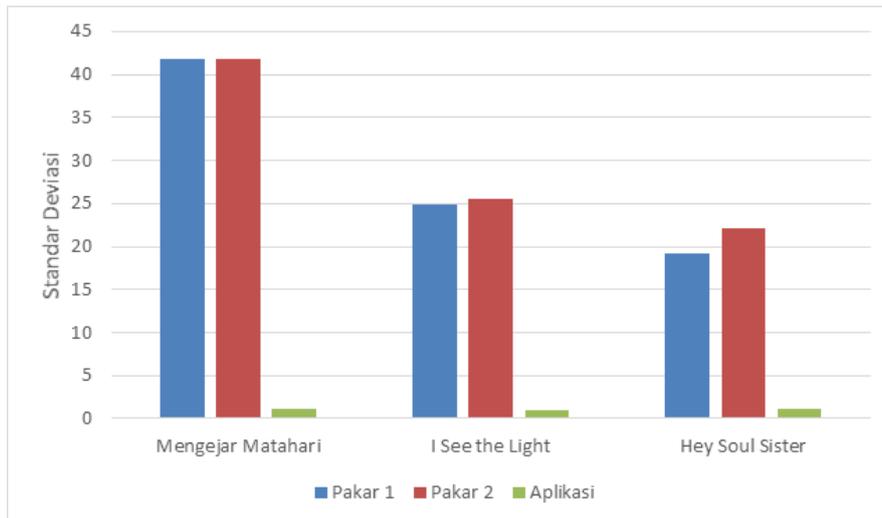
Gambar 4. Contoh antarmuka hasil distribusi

B. Pembahasan

Terdapat dua jenis pengujian yang dilakukan pada hasil implementasi, yaitu pengujian fungsionalitas dan pengujian kinerja aplikasi. Pengujian fungsionalitas adalah pengujian yang dilakukan untuk memastikan kesesuaian fungsional antara rancangan sistem dengan implementasi sistem. Dari hasil pengujian, didapatkan hasil seluruh kebutuhan fungsional telah terpenuhi, sehingga dapat dikatakan bahwa aplikasi dapat digunakan dengan wajar. Pengujian kinerja aplikasi adalah pengujian yang dilakukan untuk mengukur kualitas distribusi yang dihasilkan oleh aplikasi. Pengujian ini dilakukan secara kuantitatif dan kualitatif.

Pada pengujian kinerja secara kuantitatif, aplikasi diuji dengan membandingkan hasil distribusi luaran aplikasi dengan hasil distribusi yang diproses secara manual oleh pakar yang telah berpengalaman minimal dua tahun dalam melakukan distribusi angklung. Percobaan distribusi dilakukan pada tiga

partitur lagu, yaitu Mengejar Matahari, *I See the Light*, dan *Hey Soul Sister*. Setelah itu, pada masing-masing hasil distribusi dilakukan pendataan jumlah ketukan setiap pemain serta dihitung standar deviasinya. Besarnya standar deviasi merepresentasikan ketidakseimbangan pada distribusi tersebut. Dari pendataan, seperti yang terlihat pada Gambar 5, didapatkan hasil bahwa standar deviasi pada distribusi yang dihasilkan aplikasi jauh lebih kecil dibandingkan standar deviasi pada distribusi yang dihasilkan pakar. Oleh karena itu, dapat dikatakan bahwa aplikasi mampu menghasilkan distribusi yang lebih seimbang.



Gambar 5. Grafik Perbandingan standar deviasi hasil distribusi manual dan hasil distribusi aplikasi

Selanjutnya, untuk pengujian secara kualitatif, pengujian dilakukan dengan cara meminta pendapat beberapa pakar yang paham mengenai musik dan berpengalaman minimal dua tahun dalam melatih tim angklung untuk menilai kualitas distribusi yang dihasilkan aplikasi melalui kuisioner. Dari hasil kuisioner, pakar-pakar berpendapat bahwa aplikasi membuat proses distribusi menjadi lebih efisien dibandingkan dengan cara manual. Distribusi yang dihasilkan oleh aplikasi pun sudah cukup memuaskan karena setiap pemain mendapatkan jumlah ketukan yang cukup seimbang. Hasil distribusi juga mampu mempermudah pemain dan pelatih karena sudah langsung diberi tanda pada bagian pemain harus bermain dengan menghindari bentrok sehingga pemain dan pelatih tidak perlu memikirkan permasalahan bentrok saat latihan. Namun, terdapat beberapa poin kritik dari pakar:

1. Meskipun jumlah angklung yang dipegang memang tidak melebihi jumlah maksimum angklung setiap pemain yang menjadi masukan, akan lebih baik apabila ukuran angklung diperhatikan dalam pembagiannya. Ukuran yang terlalu mirip ataupun terlalu berbeda besar kecilnya akan mempengaruhi kenyamanan pemain saat memainkannya.
2. Setiap nomor angklung memiliki volume suara yang berbeda-beda. Pada umumnya, semakin besar nomor angklung, maka semakin kecil suaranya. Oleh karena itu, sebaiknya keberagaman volume suara ini menjadi parameter dalam proses distribusi.
3. Pewarnaan giliran main akan lebih baik menggunakan *highlight* agar terlihat lebih jelas. Namun, hal ini belum dapat dilakukan karena *Microsoft Excel* tidak memiliki fungsi untuk memberi *highlight* pada sebagian teks yang ada pada satu cell.

KESIMPULAN

Terdapat beberapa hal yang dapat disimpulkan dalam penelitian ini. Pertama, didapatkan kesimpulan bahwa distribusi giliran main angklung dapat disusun menggunakan teknik pewarnaan simpul graf dengan algoritma *Welch Powell* dan algoritma *Depth First Search* dengan cara menjadikan giliran main sebagai simpul graf dan kebentrokan sebagai sisi, menambah batasan maksimal jumlah nada (angklung) di setiap kelompok warna pada proses pewarnaan simpul graf, dan membuat proses pada algoritma *Depth First Search* menjadi memindahkan giliran main dari kelompok dengan jumlah ketukan terbanyak ke kelompok lain hingga didapatkan standar deviasi jumlah ketukan di setiap kelompok kurang dari satu atau sudah mencapai minimal dan jumlah nada (angklung) di setiap kelompok tidak melebihi jumlah yang ditentukan di awal. Selanjutnya, pengujian aplikasi memberikan hasil bahwa aplikasi mampu membuat

giliran main antarpemain menjadi lebih seimbang dan dapat membuat proses distribusi menjadi lebih efisien. Akan tetapi, ada beberapa aspek yang belum dipertimbangkan dalam aplikasi tersebut seperti keberagaman volume suara dan ukuran angklung. Terdapat beberapa saran yang dapat diberikan oleh penulis untuk penelitian selanjutnya yang sejenis. Saran pertama yaitu menambahkan parameter jumlah angklung yang dimiliki sebuah tim sehingga pada distribusi yang dihasilkan, jumlah angklung yang dibutuhkan tidak melebihi jumlah angklung yang tersedia. Selanjutnya, proses distribusi sebaiknya memperhatikan juga keberagaman ukuran dan volume suara setiap angklung. Terakhir, sebaiknya mencari media selain *Microsoft Excel* untuk menampilkan partitur hasil distribusi agar dapat menggunakan *highlight* untuk memberi tanda pada giliran main sehingga tanda tersebut dapat terlihat lebih jelas.

REFERENSI

- [1] Kemendikbud, "Angklung. Direktorat Warisan dan Diplomasi Budaya," 2015. [Online], <https://kebudayaan.kemdikbud.go.id/ditwdb/angklung/>. [Accessed October 18, 2020].
- [2] Angklungkita, "Connects Angklung to Everyone," 2018. [Online], <https://angklungkita.com/>. [Accessed October 18, 2020].
- [3] Bumi, T.P. "Aplikasi Pembangkit Distribusi Angklung dengan Teknik Fuzzy Multi-Criteria Decision Making," Tugas Akhir S1, Teknik Informatika, Institut Teknologi Bandung, Bandung, 2015.
- [4] Munir, Rinaldi, *Matematika Diskrit Revisi Keenam*, Bandung: Informatika, 2016.
- [5] Pasnur, "Implementasi Algoritma Welch-Powell dalam Pembuatan Jadwal Ujian Akhir Semester," *Jurnal Inspiration*, Jilid 2, No. 1, pp. 35-44, 2012.
- [6] Hidayah, M.R. and Prasetyo, B., "Penggunaan Metode Depth First Search (DFS) dan Breadth First Search (BFS) pada Strategi Game Kamen Rider Decade Versi 0.3.," *Scientific Journal of Informatics*, Jilid 1, No. 2, pp. 161-167, 2014.
- [7] Aslan, M. and Baykan, N.A., "A Performance Comparison of Graph Coloring Algorithms," in *International Conference on Advanced Technology & Sciences (ICAT'16)*, Selcuk University Turki, 2016. pp 266-273.
- [8] Kania, R.A.P., "Model Matematika Dalam Menentukan Distribusi Angklung Pada Suatu Penampilan," Tugas Akhir S1, Teknik Informatika, Institut Teknologi Bandung, Bandung, 2011.
- [9] Inggiantowi, H. "Sistem Tonjur untuk Membantu Menentukan Pasangan Main Angklung ke Pemain dengan Memanfaatkan MusicXML," Tugas Akhir S1, Teknik Informatika, Institut Teknologi Bandung, Bandung, 2011.