Perbandingan dan Analisis *True Random Number Generation* terhadap *Pseudorandom Number Generation* dalam Berbagai Bidang

Kevin Leonardo Handoyo/13509019
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13509019@std.stei.itb.ac.id

***Abstrak* — Pembangkit bilangan acak atau *random number generator* adalah suatu cara untuk menghasilkan sebuah bilangan acak. Bilangan acak merupakan sesuatu yang sangat penting dalam banyak bidang seperti pembangkitan kunci enkripsi dalam kriptografi, pemodelan dan simulasi berbagai fenomena, permainan-permainan dan undian/lotere, bahkan dalam bidang seni dan literatur. Namun, dengan menggunakan komputer, tidak dapat dihasilkan sebuah bilangan acak yang benar-benar acak. Bilangan acak yang dihasilkan oleh komputer merupakan bilangan semi acak atau disebut sebagai *pseudorandom number*. Sebuah komputer hanya dapat membangkitkan bilangan yang benar-benar acak dengan memanfaatkan fenomena fisik. Dalam makalah ini, akan dibahas bagaimana peran sebuah bilangan acak dalam berbagai bidang seperti permainan, lotere, seni, serta kriptografi. Dalam makalah ini juga akan dilakukan analisis dan perbandingan apabila hal-hal tersebut dilakukan dengan menggunakan pembangkit bilangan acak sebenarnya dibandingkan dengan menggunakan pembangkit bilangan acak semu.**

***Kata Kunci*— Bilangan acak, Pembangkit Bilangan Acak Semu, Pembangkit Bilangan Acak Sejati.**

I. PENDAHULUAN

Seiring dengan perkembangan teknologi informasi, berbagai hal pada jaman sekarang dilakukan dengan menggunakan komputer. Permainan komputer, pengundian lotere, pembuatan gambar, simulasi, dan sebagainya juga semakin marak dilakukan dengan menggunakan komputer. Semua hal tersebut sangatlah membutuhkan bilangan-bilangan acak.

Demikian pula dalam bidang kriptografi. Dalam kriptografi, pembangkitan sebuah kunci dalam suatu metode enkripsi, pembangkitan *Initialization Vector*, dan sebagainya sangatlah mengandalkan pembangkitan bilangan acak agar tidak mudah diserang oleh pihak yang tidak bertanggungjawab.

Bilangan acak sendiri idealnya merupakan bilangan yang tidak bisa diprediksi kemunculannya. Dari sebuah kelompok bilangan, kemungkinan bagi sebuah bilangan untuk dapat terpilih adalah sama besarnya. Dalam sebuah sekuens bilangan acak, setiap bilangan yang diambil juga idealnya tidak memiliki ketergantungan dengan bilangan-bilangan lainnya. Sifat-sifat ini sangat penting dalam banyak bidang, seperti simulasi dan kriptografi. Semakin mendekati sifat-sifat tersebut maka akan semakin baik suatu sistem simulasi maupun kemananan sebuah enkripsi.

Sayangnya, pembangkitan bilangan acak adalah sebuah hal yang sangat sulit dilakukan oleh komputer. Komputer adalah sesuatu yang sangat logis. Ia hanya dapat mengikuti sesuatu yang telah diprogram sebelumnya, karena itulah, ia tidak mungkin dapat menghasilkan sesuatu yang diluar dugaan yang berarti tidak mungkin menghasilkan sebuah bilangan acak.

Dalam menghasilkan sebuah bilangan acak, sebuah komputer dapat memakai dua pendekatan, yaitu dengan cara menggunakan pembangkit bilangan acak semu ataupun dengan menggunakan pembangkit bilangan acak sejati.

Pembangkit bilangan acak semu adalah pembangkit bilangan acak yang memanfaatkan rumus-rumus matematis dalam membangkitkan sebuah bilangan acak. Pembangkit bilangan acak semu biasanya bersifat efisien dan deterministik, namun sayangnya juga bersifat periodik.

Berbeda dengan pembangkit bilangan acak semu, pembangkit bilangan acak sejati merupakan pembangkit bilangan acak yang memanfaatkan fenomena-fenomena fisik yang kemudian ditangkap oleh komputer. Fenomena-fenomena fisik tersebut sangat beragam, mulai dari pergerakan tetikus, selisih waktu seseorang menekan tombol papan ketik, jumlah klik dalam waktu tertentu, frekuensi suara pada mikrofon, ataupun fenomena-fenomena alam lainnya yang dapat menghasilkan keacakan yang lebih baik lagi seperti peluruhan radioaktif dan kebisingan atmosfir.

Karena masing-masing pendekatan tersebut memiliki sifat masing-masing, maka akan dilakukan perbandingan serta analisis terhadap kedua metode pembangkitan bilangan acak tersebut dalam berbagai bidang, dikaitkan dengan kelemahan dan kelebihan dari masing-masing metode pembangkitan bilangan acak tersebut.

Sebagian besar percobaan yang dilakukan dalam makalah ini tidak dibuat sendiri melainkan didasarkan pada percobaan-percobaan yang telah dilakukan oleh

orang lain sebelumnya.

II. DASAR TEORI

A. Bilangan Acak

Bilangan acak adalah sebuah bilangan yang dipilih seakan-akan secara kebetulan dari beberapa sebaran spesifik sehingga sebuah kumpulan besar dari bilangan-bilangan yang terpilih tersebut dapat menyusun kembali sebaran dasarnya. Bilangan-bilangan acak tersebut juga seringkali diperlukan untuk saling independen, sehingga tidak ada hubungan antara setiap bilangan-bilangan yang berurutan.

Pembangkitan sebuah bilangan acak dapat dilakukan dengan bermacam-macam cara. Pada jaman sebelum penggunaan komputer merupakan sesuatu yang umum, sebuah bilangan acak diperoleh dengan beberapa cara, seperti pelemparan dadu, pengocokan kartu, pembacaan tabel bilangan random, dan lain sebagainya. Ketika komputer sudah mulai biasa digunakan, yaitu sekitar tahun 1940 hingga sekarang, sebuah bilangan acak pada umumnya dibangkitkan secara numerik atau aritmatik dengan menggunakan komputer. Bilangan acak yang berasal dari pembangkitan bilangan acak dengan metode tersebut disebut sebagai bilangan acak semu.

Tidaklah mungkin untuk membuat sebuah *string* yang panjang dari bilangan-bilangan satuan yang acak dan membuktikan keacakan dari *string* tersebut. Anehnya, adalah sangat sulit juga bagi manusia untuk menghasilkan *string* yang panjang yang berisi *digit-digit* bilangan yang acak. Bahkan, program komputer dapat dibuat untuk memprediksi *digit* yang akan dipilih berikutnya oleh manusia. Hal ini dikarenakan setiap manusia memiliki kecenderungan dalam mengambil suatu keputusan, sehingga dapat tampak sebuah pola yang sering dipilih oleh manusia tersebut.

Pembangkitan bilangan acak dapat dibedakan menjadi dua kelompok besar, yaitu pembangkitan bilangan acak semu dengan menggunakan rumus-rumus matematika, dan pembangkitan bilangan acak sejati, yaitu dengan menangkap fenomena-fenomena alam pada komputer. Dalam pembangkitan acak semu dan sejati tersebut, masing-masing juga memiliki sangat banyak variasi dalam menghasilkan sebuah bilangan acak. Setiap variasi tersebut memiliki tingkat keacakan yang berbeda satu sama lainnya.

B. Pembangkit Bilangan Acak Semu

Pembangkit bilangan acak semu (*pseudorandom number generator*) adalah suatu pembangkit bilangan acak yang berdasarkan pada rumus-rumus matematika. Oleh karena menggunakan rumus-rumus matematis, perbedaan formula yang digunakan dalam algoritmanya akan menyebabkan banyak sekali variasi dari pembangkit bilangan acak semu yang telah dibuat. Pembangkit bilangan acak semu merupakan sebuah algoritma yang berfungsi membangkitkan bilangan-bilangan yang memiliki sifat yang mendekati sifat bilangan acak yang sebenarnya. Sesuai dengan namanya, bilangan-bilangan yang dihasilkan oleh pembangkit bilangan acak semu sebenarnya tidak benar-benar acak. BIlangan-bilangan ini ditentukan oleh sebuah himpunan bilangan yang relatif kecil yang digunakan sebagai sebuah nilai awal, dimana sebuah *seed* yang acak juga termasuk sebagai bagian dari nilai awal tersebut.

Pembangkit blangan acak semu yang berbasis komputer pertama kali ditemukan oleh John von Neumann sekitar tahun 1940-an. Ia membuat sebuah algoritma pembangkitan bilangan acak semu yang bernama algoritma middle-square. Ide dari algoritma ini adalah dengan mengambil sebuah angka yang acak, kemudian angka itu dikuadratkan dan *digit* yang beradadi tengah dari hasilnya itu diambil sebagai bilangan acak yang kemudian dijadikaan sebagai *seed* pada iterasi pengambilan bilangan acak berikutnya. Pada algoritma ini, terdapat sebuah masalah, yaitu urutan-urutan bilngan acaak tersebut akan terus berulang. Von Neumann menyadari hal ini, namun ia tidak membuat perbaikan dari algoritma ini karena ia menganggap bahwa pembangkitan bilangan acak dengan metode ini sudah menjawab kebutuhannya, selain itu ia khawatir bahwa perbaikan secara matematis hanya menyebabkan penyembunyian kesalahan yang terjadi, bukan memperbaikinya.

Sebuah pembangkit bilangan acak semu serta bilangan acak yang dihasilkannya memiliki sifat-sifat sebagai berikut:

1. Efisien
   Artinya, sebuah pembangkit bilangan acak semu dapat menghasilkan banyak bilangan acak dalam waktu yang singkat.
2. Deterministik
   Artinya, sebuah bilangan acak dapat dibangkitkan kembali pada waktu yang akan datang apabila parameter-parameter awal yang digunakan dalam pembangkitan bilangan tersebut diketahui.
3. Periodik
   Artinya, sekuens-sekuens yang terbentuk pada bilangan acak yang dihasilkan akan mengulangi dirinya sendiri dalam suatu periode tertentu.

Setiap kali pembangkitan bilangan acak oleh pembangkit bilangan acak semu dilakukan, selalu dibutuhkan sebuah *state* awal yang disebut sebagai *seed state*. Apabila *seed state* yang digunakan adalah sama, maka hasil yang dihasilkan oleh pembangkit bilangan acak semua juga selalu sama. Periode dari sebuah pembangkit bilangan acak semu adalah panjang dari *digit-digit* bilangan yang tidak berulang. Panjang periode suatu pembangkit bilangan acak semu sangat tergantung kepada ukuran *seed state* yang dihitung dalam bit. Semakin panjang ukuran suatu *seed state*, maka periode juga akan semakin panjang dan bilangan acak yang dihasilkan akan semakin baik.

Beberapa masalah yang dimiliki oleh pembangkit bilangan acak semu adalah periode yang lebih pendek dari harapan untuk beberapa *seed state* tertentu, kurang seragamnya distribusi untuk angka yang dihasilkan dalam jumlah besar, adanya keterkaitan antara bilangan yang

berurutan, serta adanya kemungkinan bahwa beberapa angka tidak akan pernah keluar.

Beberapa contoh algoritma yang digunakan dalam pembangkit bilangan acak semu antara lain *linear congruential generator*, *multiplicative random number generator*, serta *mixed congruential random number generator*.

Linear Congruential Generator memiliki rumus sebagai berikut:

$$Z_i = (aZ_{i-1} + c) \bmod m$$

dimana:

Z_i = bilangan acak ke-i dari deretnya
Z_{i-1} = bilangan acak sebelumnya
a = faktor pengali
c = faktor penjumlah
m = faktor modulus

Metode ini membutuhkan sebuah *seed* awal yaitu Z_0. Dengan menggunakan metode ini, periode pengulangan bilangan tidak akan lebih besar dari m. Metode ini akan mempunyai periode penuh, yaitu sebesar m-1 apabila syarat-syarat berikut dipenuhi:

1. c relatif prima terhadap m
2. a-1 dapat dibagi dengan semua faktor prima dari m
3. a-1 adalah kelipatan 4 jika m adalah kelipatan 4
4. m > max(a,c,Z_0)
5. a>0, c>0

Karena itulah, algoritma ini sangat dipengaruhi oleh penentuan konstanta-konstantanya, yaitu a,c dan m, untuk memperoleh hasil yang memiliki periode sepanjang yang diinginkan.

Algoritma *Multiplicative Random Number Generator* mengikuti rumus berikut:

$$Z_i = (a.Z_{i-1}) \bmod m$$

dimana:

Z_i = bilangan acak ke-i dari deretnya
Z_{i-1} = bilangan acak sebelumnya
a = faktor pengali
m = faktor modulus

Agar nilai yang dihasilkan semakin acak, maka dapat mengikuti ketentuan-ketentuan berikut:

1. Nilai m dipilih sebesar mungkin agar periode semakin besar, selain itu diusahakan sepanjang suatu kata yang digunakan pada komputer tersebut.
2. Nilai a dipilih agar korelasi antar Z_n minimum, yaitu dengan menggunakan nilai a yang berupa bilangan ganjil dan relatif prima terhadap m.
3. *Seed* Z_0 berupa bilangan ganjil yang sebesar mungkin, namun masih tetap lebih kecil daripada m.

Rumus yang digunakan dalam algoritma *Mixed Congruential Random Number Generator* adalah sebagai berikut:

$$Z_n = a^n Z_0 + \frac{a^n - 1}{a - 1}.C(mod.m)$$

Beberapa kondisi yang harus dipenuhi apabila menggunakan algoritma ini adalah sebagai berikut:

1. c merupakan bilangan yang relatif prima terhadap nilai n.
2. a= 1 (mod q) untuk setiap faktor prima q dari m
3. a = 1 (mod 4) apabila 4 adalah suatu faktor dari m

Kondisi kedua berarti

$$a - q\left(\frac{a}{q}\right) = 1$$

$$k = \left(\frac{a}{q}\right)$$

apabila akan dapat diperoleh untuk a, yaitu a=1+qk dimana q adalah faktor prima dari m.

C. Pembangkit Bilangan Acak Sejati

Pembangkit bilangan acak sejati adalah pembangkit bilangan acak yang mendapatkan keacakanannya dari fenomena fisik dan memasukannya ke dalam komputer. Fenomena fisik yang dimaksud sangatlah bermacam-macam, mulai dari yang paling sederhana seperti pergerakan tetikus, sampai peluruhan radioaktif.

Beberapa sumber pembangkit bilangan acak sejati:

- Sumber radioaktif
- efek quantum pada semikonduktor
- polarisasi foton
- mikrofon
- video kamera
- jeda waktu antar ketikan
- pergerakan tetikus

Langkah-langkah untuk memperoleh bit-bit random adalah sebagai berikut:

1. Mendapatkan bit-bit angka

   Pertama-tama seseorang mengumpulkan beberapa bit yang tidak diketahui dan tidak dapat ditebak oleh pihak lawan. Agnka-angka ini harus berasal dari sebuah perangkat I/O. Bit-bit tersebut tidak harus semuanya saling independen. Artinya, seseorang dapat memprediksi sebagian angkanya dengan kemungkinan lebih dari 0.5 apabila diketahui bit-bit lainnya. Yang penting disini adalah bit-bit ini menyimpan informasi (entropi) yang tidak diketahui oleh pihak lawan.
2. Menentukan entropi

   Langkah berikutnya adalah untuk menentukan berapa banyak bit yang tidak dapat terduga yang diperoleh. Artinya, seseorang harus mengetahui berapa banyak dari bit yang diperoleh adalah independen dan tidak dapat ditebak. Jumlah bit-bit ini biasanya disebut sebagai entropi.
3. Mereduksi sampai ke bit-bit independen

   Dalam langkah ini, seseorang dapat menghitung *hash* dari bit-bit yang diperoleh untuk mereduksi mereka menjadi bit-bit yang benar-benar acak dan independen. Fungsi *hash* pada tahap ini harus memiliki setiap bit keluaran bergantung secara fungsional kepada setiap bit masukan dan independen secara fungsional terhadap bit-bit keluaran lainnya. Jenis-jenis fungsi *hash* yang cocok untuk melakukan hal ini adalah yang baik secara kriptografi, misalnya MD5 dan SHA. Hasil

dari langkah ketiga ini adalah sebuah himpunan dari bit-bit yang independen dan tidak dapat ditebak.

III. Penerapan Bilangan Acak dalam Berbagai Bidang

A. Permainan

Dalam sebuah permainan, khususnya dalam permainan yang berupa permainan-permainan kasino, sebuah bilangan acak memiliki peran yang sangat penting. Dalam permainan berjenis *roleplaying game*, permainan strategi, dan yang berjenis *action*, sebuah bilangan acak dapat digunakan untuk menentukan keberhasilan kita dalam melakukan sesuatu, misalnya saat akan menempa senjata, mendapatkan barang dari sebuah peti, sampai ke pilihan aksi yang akan dilakukan oleh musuh kita.

Dalam genre lainnya, seperti permainan berbasis kasino ataupun *board games*, peran bilangan acak akan lebih nyata lagi. Dalam permainan *roullete* misalnya, pemilihan angka acak bagi pemenangnya sangatlah penting. Selain itu, dalam *board game* seperti permainan-permainan kartu, bilangan acak dapat dimanfaatkan untuk penyusunan dek sehingga komputer tidak harus menyimpan setiap *state* dek yang mungkin, serta dapat digunakan bagi komputer untuk menentukan langkah mana yang akan dipilih saat akan menjalankan suatu langkah disaat semua langkah merupakan langkah yang valid.

B. Lotere dan Undian

Dalam bidang lotere dan undian, semuanya bergantung kepada bilangan acak. Semua pemenang ditentukan oleh sebuah bilangan yang dipilih secara benar-benar acak. Selain dipakai untuk menentukan pemenang, jaman sekarang juga sudah sangat banyak dibuat sebuah pembangkit bilangan acak yang ditujukan terhadap para pemain lotere dan undian untuk membantu mereka dalam menentukan bilangan yang akan dipasang dalam sebuah lotere.

Bilangan-bilangan acak yang digunakan dalam lotere dan undian ini diharapkan memiliki beberapa kualitas, seperti tidak terjadinya pengulangan, serta kemungkinan yang sama bagi setiap angka untuk dapat muncul, sehingga tidak ada angka yang tidak mungkin bisa memenangkan sebuah lotere.

C. Statistika

Dalam statistika, penggunaan bilangan acak sangat penting ketika kita akan melakukan *random sampling*. *Random sampling* adalah sebuah pemilihan beberapa sampel yang dipilih secara acak dari sebuah populasi tertentu.

Misalnya ada 100 orang pada suatu kotadan kita akan memilih 10 orang secara acak. Orang-orang tersebut kemudian dapat kita beri angka dari 001 sampai 100. Kemudian, kita dapat mengambil 10 orang secara acak dengan menggunakan berbagai cara, seperti mengacu kepada tabel bilangan acak, menggunakan pembangkit bilangan acak semu, ataupun pembangkit bilangan acak sejati.

Agar pengambilan sampel dapat mewakili populasi, tidak boleh ada pola tertentu dalam pengambilan sampel tersebut sehingga periode harus tidak terdeteksi dan sebisa mungkin semuanya independen.

D. Simulasi dan Pemodelan

Dalam bidang simulasi dan pemodelan, bilangan-bilangan acak sangatlah diperlukan agar dapat melakukan simulasi yang benar-benar baik. Sebuah simulasi dan model harus dapat menerima segala macam masukan agar memiliki kualitas yang lebih baik. Apabila suatu model atau aplikasi untuk simulasi menerima sesuatu yang terasa acak namun hanya tertentu saja yang mungkin muncul, hal ini dapat membahayakan karena simulasi yang dilakukan dapat menjadi tidak lengkap sehingga tidak valid.

Kualitas bilangan acak yang dikehendaki dalam bidang simulasi dan pemodelan adalah bilangan acak yang pembangkitannya efisien, karena besarnya jumlah data dan bilangan acak yang akan dibutuhkan, serta hampir tidak memiliki periode sehingga tidak ada tes yang terus menerus berulang.

E. Seni

Dalam bidang seni, sebuah bilangan acak dapat digunakan sebagai penambahan elemen kreatif di dalam sebuah seni itu. Misalnya pada bidang musik, terkadang dibutuhkan suatu permainan alat musik yang acak untuk menambah nilai seni dari musik itu sendiri, begitu pula yang terjadi pada sebuah gambar. Jaman sekarang ini, karya lukisan-lukisan abstrak juga semakin banyak peminatnya, sehingga sebuah bilangan acak dapat membantu seorang seniman dalam membuat suatu gambar yang terkesan acak.

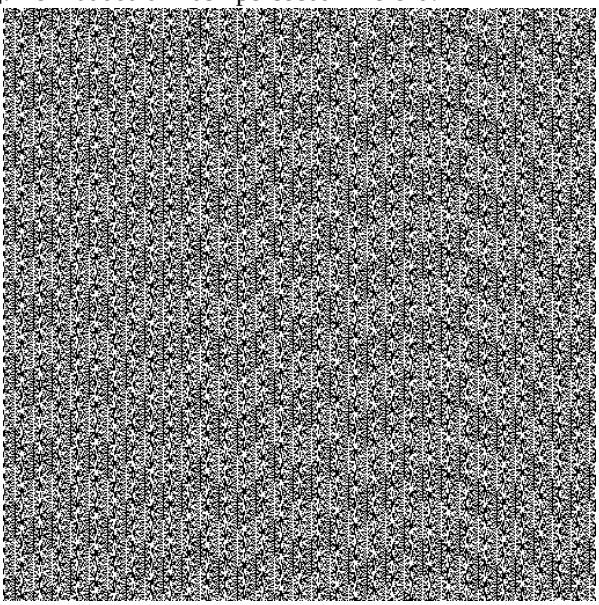
F. Kriptografi

Dalam bidang kriptografi, pembangkitan bilangan acak sangat penting dalam pembangkitan kunci yang digunakan untuk enkripsi, misalnya pada *keystream generator* yang digunakan sebagai kunci dalam algoritma cipher aliran, ataupun sebagai pembangkit *Initialization Vector* yang digunakan dalam cipher blok dengan metode CFB.

Bilangan acak dalam bidang kriptografi dikehendaki agar bersifat saling independen, sehingga apabila pihak lawan mengetahui sebagian kunci, ia masih tidak akan bisa menebak kunci secara utuh. Kunci juga idealnya tidak memiliki periode. Kalaupun sulit untuk mencapai hal ini, periode dari suatu bilangan acak yang digunakan diusahakan agar mencapai nilai yang sangat besar sehingga akan menyulitkan pihak lawan untuk mendapatkan kuncinya.

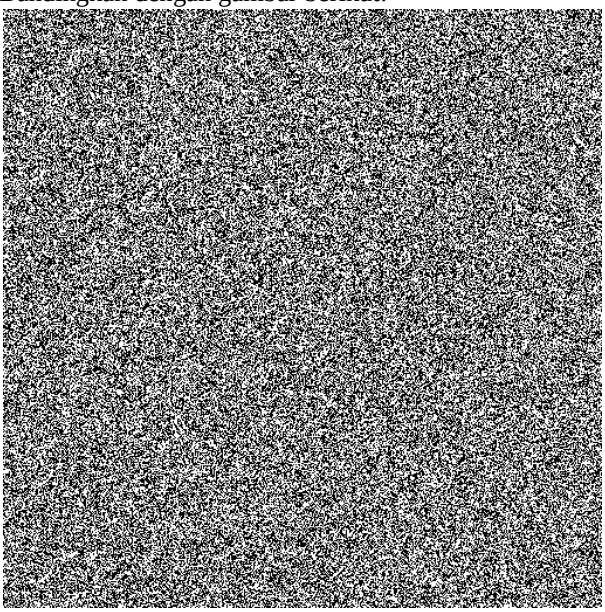
IV. ANALISIS

Dalam membuktikan sifat-sifat dari kedua pembangkit bilangan acak yang telah dibahas sebelumnya, kita dapat melihat pada beberapa percobaan dengan menggunakan pembangkitan sebuah gambar *bitmap* acak yang telah dilakukan oleh Bo Allen dan John Ramey pada [5] dan [6]. Berikut adalah hasil percobaan mereka:



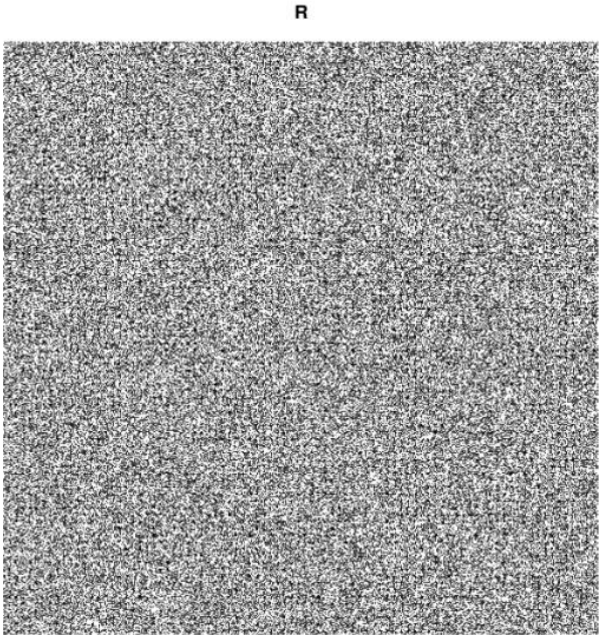
Gambar 1. Hasil gambar bitmap acak dengan menggunakan fungsi rand() pada bahasa php...[5]

Bandingkan dengan gambar berikut:



Gambar 2. Hasil gambar bitmap acak dengan memanfaatkan kakas pada situs random.org (menggunakan *True Random Number Generator*)...[5]

Dari kedua gambar tersebut dapat terlihat bahwa pembuatan gambar bitmap dengan menggunakan pembangkit bilangan acak semu yang sederhana seperti yang ada dalam bahasa php menimbulkan pola yang dapat terlihat dengan jelas, sedangkan pada hasil gambar menggunakan pembangkit bilangan acak sejati, tidak tampak pola apapun. Namun, itu tidak berarti pembangkit bilangan acak semu sangat buruk, sebagai contoh kita lihat hasil berikut:



Gambar 3. Hasil gambar bitmap acak dengan menggunakan fungsi random buatan John Ramey dalam bahasa R...[6]

Dari contoh ini, tidak nampak adanya pola seperti yang terlihat pada gambar 1 walaupun sama-sama menggunakan pembangkit bilangan acak semu. Selain itu, mari kita lihat metode LCG pada pembangkit bilangan acak semu sebagai contoh, yang telah dilakukan oleh teman saya Adriano Milyardi dalam makalahnya tentang Vide Noir Number[9], dengan parameter masukan sebagai berikut:

a=2
c=3
m=97
n=100
Z_0=0

hasil yang diperoleh adalah sebagai berikut:

3	82	21	46
9	70	45	95
21	46	93	96
45	95	92	1
93	96	90	5
92	1	86	13
90	5	78	29
86	13	62	61
78	29	30	28
62	61	63	59
30	28	32	24

63	59	67	51
32	24	40	8
67	51	83	19
40	8	72	41
83	19	50	85
72	41	6	76
50	85	15	58
6	76	33	22
15	58	69	47
33	22	44	0
69	47	91	3
44	0	88	9
91	3	82	21
88	9	70	45

Dari data diatas, tampak bahwa pada digit ke 97, pola kembali berulang. Selain itu, hanya ada kemungkinan 48 jenis bilangan yang muncul dari yang seharusnya sebanyak 97 bilangan yang mungkin muncul.

Selain itu, bila kita tinjau dari berbagai bidang, tergantung dengan apa kebutuhan dari bidang-bidang tersebut akan bilangan acak, dapat ditentukan mana yang lebih cocok untuk digunakan, pembangkit bilangan acak semu atau sejati.

Dalam bidang permainan, pembangkitan bilangan acak pada umumnya tidak terlalu sering dilakukan, meskipun ada beberapa permainan yang membutuhkan bilangan acak yang banyak untuk dibangkitkan, namun jumlahnya tidak terlalu besar dan waktunya juga tidak harus terlalu cepat. Selain itu, agar sebuah permainan menjadi menarik, diperlukan tingkat keacakan yang tinggi agar pemain tidak dapat mencari pola-pola langkah yang dipilih AI. Karena itulah dapat terlihat bahwa pembangkit bilangan acak sejati lebih cocok untuk digunakan dalam bidang permainan.

Demikian juga dalam bidan lotere dan undian, serta bidang statistika. Seperti yang telah dibahas pada bagian sebelumnya, kebutuhan akan bilangan acak pada kedua bidang tersebut sesuai dengan sifat-sifat pembangkit bilangan acak sejati sehingga pembangkit bilangan acak sejati lebih cocok untuk digunakan dalam kedua bidang ini.

Selain itu, dalam bidang kriptografi, kita dapat mengambil contoh *keystream generator* pada cipher aliran. Dengan menggunakan algoritma sederhana dimana sebuah kunci dibangkitkan dengan sebuah *seed* n bit yang kemudian untuk bit-bit selanjutnya pada kunci merupakan hasil XOR dari bit pertama dan terakhir dari n bit sebelumnya, akan terjadi pengulangan setiap 2^n-1 bit sekali. Pengulangan ini tentunya tidak dikehendaki dalam kriptografi, sedangkan semua algoritma pembangkitan bilangan acak semu akan mengalami pengulangan, walaupun telah ada beberapa algoritma yang sangat baik dimana periode pengulangannya sangat besar, namun tetap saja masih ada masalah dimana setiap bitnya masih bergantung satu sama lain. Karena itulah, pembangkit bilangan acak sejati juga lebih tepat digunakan dalam bidang kriptografi.

Penentuan yang paling sulit dilakukan adalah dalam bidang simulasi. Di satu sisi, ia membutuhkan efisiensi yang sangat tinggi karena jumlah data yang dibutuhkan sangatlah banyak. Namun, di lain sisi, ia juga membutuhkan bilangan-bilangan yang benar-benar acak agar simulasinya semakin baik dan nyata. Selain itu, terkadang dibutuhkan juga simulasi terhadap suatu nilai tertentu yang harus diulang, sehingga kadang diperlukan juga sifat deterministik dari bilangan acak semu. Karena itulah, pada bidang simulasi tidak dapat ditentukan yang mana yang lebih tepat karena tergantung pada kebutuhan dari simulasi itu sendiri.

Demikian pula dalam bidang seni. Seperti yang telah diperlihatkan pada bagian awal analisis, penggunaan fungsi pembangkit bilangan acak semu yang buruk dapat menghasilkan pola yang berulang, sedangkan pembangkit bilangan acak sejati akan menghasilkan gambar yang benar-benar acak dan tanpa pola. Namun,. dalam bidang seni, terkadang dibutuhkan juga pola-pola semacam gambar 1, dan terkadang yang diinginkan adalah pola yang benar-benar acak. Karena itulah, dalam bidang seni, metode mana yang dipilih sangat bergantung pada hasil yang diharapkan.

V. KESIMPULAN

Berdasarkan analisis yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut:

- Pembangkit bilangan acak sejati sebenarnya lebih dibutuhkan, namun membutuhkan perangkat keras yang cenderung sulit didapatkan oleh orang pada umumnya. Selain itu, pembangkit bilangan acak ini juga tidak efisien.
- Pembangkit bilangan acak semu dapat menyerupai pembangkit bilangan acak sejati apabila algoritma yang digunakan cukup kompleks dan panjang *seed* sangat panjang.
- Dalam bidang kriptografi, lebih cocok digunakan pembangkit bilangan acak sejati.
- Dalam bidang permainan, lebih cocok digunakan pembangkit bilangan acak sejati.
- Dalam bidang lotere dan undian, lebih cocok digunakan pembangkit bilangan acak sejati.
- Dalam bidang statistika, lebih cocok digunakan pembangkit bilangan acak sejati.
- Dalam bidang simulasi dan pemodelan, pembangkit bilangan acak yang dipilih tergantung pada apa yang diharapkan.
- Dalam bidang seni, pembangkit bilangan acak yang dipilih tergantung pada apa yang diharapkan.

REFERENSI

[1]Munir, Rinaldi. 2005. Diktat Kuliah IF3054 Kriptografi. Departemen Teknik Informatika ITB.

[2]http://elib.unikom.ac.id/files/disk1/471/jbptunikompp-gdl-rianilubis-23527-5-06rando-r.pdf (12/5/2012)

[3]http://mathworld.wolfram.com/RandomNumber.html(13/5/2012)

[4]http://www.random.org(12/5/2012)

[5]http://boallen.com/random-numbers.html(12/5/2012)

[6]http://johnramey.net/blog/2011/11/25/pseudo-random-vs-random-numbers-in-r/(12/5/2012)

[7]http://www.bookrags.com/research/random-numbers-wsd/(13/5/2012)

[8]http://www.std.com/~cme/P1363/ranno.html(13/5/2012)

[9]Milyardi, Adriano. 2012. Vide Noir Number

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 19 Maret 2011

ttd

Kevin Leonardo Handoyo/13509019