



OPTIMASI PORTOFOLIO DENGAN KENDALA *BUY-IN THRESHOLD* MENGGUNAKAN METODE CUCKOO SEARCH

Lilik Muzdalifah¹, Kuntjoro Adji Sidarto²

¹FMIPA Universitas PGRI Ronggolawe Tuban

²FMIPA Institut Teknologi Bandung

¹moezdalifah@yahoo.co.id

²sidarto@math.itb.ac.id

ABSTRAK

Optimasi portofolio merupakan permasalahan membentuk komposisi dari berbagai aset dalam portofolio sehingga didapatkan hasil yang optimal. Penyusunan portofolio perlu memperhatikan batas bawah dari proporsi investasi dikarenakan para investor lebih memilih untuk tidak berinvestasi daripada berinvestasi dalam jumlah yang terlalu kecil (*buy-in threshold*). Optimasi portofolio dengan kendala *buy-in threshold* dirumuskan ke dalam dua jenis masalah, yaitu masalah yang hanya melibatkan variabel real/kontinu [2], dan masalah yang melibatkan variabel real-bulat [3]. Pada penelitian ini, digunakan algoritma Cuckoo Search untuk optimasi portofolio yang hanya melibatkan variabel real/kontinu. Selanjutnya, dilakukan modifikasi terhadap algoritma Cuckoo Search sehingga dapat digunakan untuk menyelesaikan masalah yang melibatkan variabel real-bulat (Mixed Integer Nonlinear Programming). Model portofolio yang digunakan adalah model Markowitz yang melibatkan masalah meminimumkan risiko (Minrisk) dan masalah memaksimumkan return (Maxret). Hasil penelitian menunjukkan bahwa pada optimasi portofolio tanpa kendala *buy-in threshold*, masih dijumpai proporsi saham yang lebih kecil dari y_{min} . Sedangkan, pada optimasi portofolio dengan kendala *buy-in threshold* sudah tidak dijumpai proporsi saham yang lebih kecil dari y_{min} . Proporsi saham tersebut menjadi lebih besar atau sama dengan y_{min} , atau sama dengan nol. Serta dapat diamati pula bahwa hasil optimasi portofolio dengan kendala *buy-in threshold* yang dikerjakan dengan metode Cuckoo Search dan modifikasi Cuckoo Search memberikan hasil yang dekat atau bisa dikatakan sama.

Kata Kunci: *optimasi portofolio, buy-in threshold, mixed integer nonlinear programming, algoritma cuckoo search*

1. Pendahuluan

Portofolio merupakan kumpulan investasi yang dimiliki oleh institusi atau perorangan. Masing-masing investasi memiliki imbal hasil (*return*) dan risiko yang berbeda. Masalah optimasi portofolio merupakan permasalahan membentuk komposisi dari berbagai aset dalam portofolio sehingga didapatkan hasil yang optimal. Markowitz [1] telah mengembangkan teori portofolio modern yang melibatkan masalah memaksimumkan *return* (Maxret) dan meminimumkan risiko (Minrisk). Suatu portofolio disebut sebagai portofolio efisien jika portofolio tersebut menghasilkan *return* maksimum dengan risiko tertentu atau menghasilkan risiko minimum dengan target *return* tertentu [1]. Dengan menghindari proporsi investasi yang bernilai negatif ($y_i < 0$), didefinisikan masalah Minrisk dan Maxret tanpa *short-selling* sebagai berikut,

$$\min V = \vec{y}^T Q \vec{y} \quad (1)$$

Kendala $\vec{r}^T \vec{y} = Rp$, $\vec{e}^T \vec{y} = 1$, dan

$$y_i \geq 0, i = 1, 2, \dots, n$$

dan

$$\min R = -\vec{r}^T \vec{y} \quad (2)$$

Kendala $\vec{y}^T Q \vec{y} = Va$, $\vec{e}^T \vec{y} = 1$, dan

$$y_i \geq 0, i = 1, 2, \dots, n$$

dimana V , R , y_i , r_i , Rp , dan Va , secara berturut-turut adalah risiko, *return*, proporsi investasi aset i , rata-rata *return* aset i , target *return*, dan risiko yang berani ditanggung. Serta,

$$Q = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \dots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \dots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \dots & \sigma_{nn} \end{pmatrix} \text{ dan } \vec{e} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$$

dengan σ_{ii} adalah variansi dari aset $_i$ dan σ_{ij} adalah kovariansi dari aset $_i$ dan aset $_k$. Masalah (1) dan (2) dapat diselesaikan dengan cara mengubahnya menjadi masalah optimasi tanpa kendala, yaitu dengan menggunakan fungsi penalti. Untuk menjaga ketaknegatifan y_i , diperkenalkan variabel baru x_i sehingga $y_i = x_i^2$, $i = 1, 2, \dots, n$. Fungsi penalti yang akan diminimumkan menjadi

$$V = \vec{y}^T Q \vec{y} + \rho(\vec{e}^T \vec{y} - 1)^2 + \rho \left(\frac{\vec{r}^T \vec{y}}{Rp} - 1 \right)^2$$

dan

$$R = -\vec{r}^T \vec{y} + \rho(\vec{e}^T \vec{y} - 1)^2 + \rho \left(\frac{\vec{y}^T Q \vec{y}}{Va} - 1 \right)^2$$

dimana ρ adalah parameter penalti yang merupakan bilangan positif yang cukup besar. Pada kenyataannya, para investor lebih memilih untuk tidak berinvestasi daripada berinvestasi dengan proporsi yang terlalu kecil, sehingga agar masalah optimasi portofolio menjadi nyata, perlu ditambahkan kendala *buy-in threshold*, yaitu kendala yang memberikan batas bawah tak nol (y_{min}) untuk setiap aset. Telah banyak penelitian yang mengkaji masalah optimasi portofolio dengan kendala *buy-in threshold*. Referensi [2] mengkaji masalah tersebut menggunakan pendekatan minimisasi global yang hanya melibatkan variabel-variabel kontinu (real). Metode yang digunakan adalah metode *Direct* yang disempurnakan dengan *Quasy-Newton*. Permasalahan Minrisk dan Maxret dengan kendala *buy-in threshold* didefinisikan sebagai masalah (1) dan (2), dengan tambahan kendala

$$\phi(y_i) = 4 \frac{y_i(y_i - y_{min})}{y_{min}^2} \geq 0 \text{ untuk } i = 1, 2, \dots, n$$

Sehingga, fungsi penalti yang akan diminimumkan menjadi

$$V = \vec{y}^T Q \vec{y} + \rho(\vec{e}^T \vec{y} - 1)^2 + \rho \left(\frac{\vec{r}^T \vec{y}}{Rp} - 1 \right)^2 + \mu \sum_{i=1}^n \psi(y_i)^2 \quad (3)$$

dan

$$R = -\vec{r}^T \vec{y} + \rho(\vec{e}^T \vec{y} - 1)^2 + \rho \left(\frac{\vec{y}^T Q \vec{y}}{Va} - 1 \right)^2 + \mu \sum_{i=1}^n \psi(y_i)^2 \quad (4)$$

dimana

$$\psi(y_i) = \min(0, \phi(y_i)), i = 1, 2, \dots, n$$

ρ dan μ adalah parameter penalti yang merupakan bilangan positif yang cukup besar. Referensi [3] mengkaji masalah optimasi portofolio dengan kendala *buy-in threshold* dengan merumuskannya sebagai masalah *Mixed Integer Nonlinear Programming* (MINLP). Permasalahan Minrisk dan Maxret dengan kendala *buy-in threshold* didefinisikan sebagai

$$\min V = \vec{y}^T Q \vec{y} \quad (5)$$

Kendala $\vec{r}^T \vec{y} = Rp$, $\vec{e}^T \vec{y} = 1$, dan

$$z_i y_{max} \geq y_i \geq z_i y_{min}, i = 1, 2, \dots, n$$

dimana $z_i \in \{0,1\}$

dan

$$\min R = -\vec{r}^T \vec{y} \quad (6)$$

Kendala $\vec{y}^T Q \vec{y} = Va$, $\vec{e}^T \vec{y} = 1$, dan

$$z_i y_{max} \geq y_i \geq z_i y_{min}, i = 1, 2, \dots, n$$

dimana $z_i \in \{0,1\}$

dimana y_{min} dan y_{max} adalah batas bawah dan batas atas proporsi investasi, dalam hal ini $y_{max} = 1$. Penelitian ini mengkaji tentang masalah optimasi portofolio dengan kendala *buy-in threshold* yang dirumuskan ke dalam dua jenis masalah, yaitu masalah yang hanya melibatkan variabel-variabel kontinu (real), seperti pada (3) dan (4), dan masalah yang melibatkan variabel bilangan bulat, seperti pada (5) dan (6). Kemudian masalah tersebut diselesaikan menggunakan metode *Cuckoo Search*, yaitu suatu metode metaheuristik yang terinspirasi dari perilaku unik burung *Cuckoo*. Metode ini pertama kali dikembangkan oleh Yang dan Deb, [4], dengan menggunakan asumsi

- Setiap burung *Cuckoo* hanya meletakkan satu telur pada satu waktu dan meletakkannya pada sarang burung lain yang dipilih secara acak.
- Sarang terbaik dengan telur berkualitas tinggi (solusi), akan lolos menuju generasi selanjutnya.
- Banyaknya sarang yang tersedia tetap, dan peluang telur burung *Cuckoo* ditemukan oleh burung yang menjadi sasaran parasitnya adalah $Pa \in [0,1]$. Jika telur burung *Cuckoo* ditemukan, maka burung pemilik sarang akan meninggalkan sarangnya dan membuat sarang baru.

Dalam hal ini, telur pada sarang burung yang menjadi sasaran parasit burung *Cuckoo* merepresentasikan suatu solusi. Telur burung *Cuckoo* yang berkualitas tinggi kemudian akan menjadi solusi baru, yang akan menggantikan solusi yang kurang baik dalam sarang tersebut. Algoritma *Cuckoo Search* tersebut dikombinasikan dengan *Lévy Flight*, yaitu random walk dengan langkah acak yang berdistribusi *Lévy*. Panjang langkah, s , didefinisikan

$$s = \frac{u}{|v|^{1/\beta}}, \quad 0 < \beta \leq 2 \quad (7)$$

$$u \sim N(0, \sigma_u), \quad v \sim N(0, \sigma_v) \quad (8)$$

$$\sigma_u = \left\{ \frac{\Gamma(1+\beta) \sin(\pi\beta/2)}{\Gamma[(1+\beta)/2] \beta 2^{(\beta-1)/2}} \right\}^{1/\beta}, \quad \sigma_v = 1 \quad (9)$$

Pada penelitian ini, masalah (3) dan (4) akan diselesaikan menggunakan algoritma *Cuckoo Search*, sedangkan masalah (5) dan (6) diselesaikan dengan melakukan suatu modifikasi pada algoritma *Cuckoo Search* sehingga algoritma yang telah dimodifikasi tersebut mampu untuk menyelesaikan masalah *Mix Integer Nonlinear Programming*.

2. Metode Penelitian

Tujuan utama dari penelitian ini adalah untuk menyelesaikan masalah optimasi portofolio dengan kendala *buy-in threshold* menggunakan metode *Cuckoo Search*. Data yang digunakan adalah data dari 20 saham LQ45 yang dipilih secara acak oleh penulis dan 20 saham pertama FTSE. Pada data 20 saham LQ45 digunakan kode saham dengan urutan seperti pada list saham

LQ45 [5]. Data 20 saham LQ45 yang diambil adalah data mingguan pada periode 7 Mei 2009-7 Mei 2014 [6], dengan *mean return*,

TABEL 1. *MEAN RETURN* 20 SAHAM LQ45

Saham	Mean Return	Saham	Mean Return
1	0.003113	17	0.007125
2	0.010852	20	0.008332
4	0.009245	21	0.006833
6	0.009891	22	0.006124
7	0.005331	23	0.002617
8	0.005827	25	0.009755
11	0.005511	31	0.003667
12	0.006026	32	0.001273
13	0.011361	41	0.004410
14	0.011594	42	0.006108

Sedangkan data 20 saham pertama FTSE yang diambil adalah data mingguan pada periode Maret 1992-September 1997 [7], dengan *mean return*,

TABEL 2. *MEAN RETURN* 20 SAHAM PERTAMA FTSE

Saham	Mean Return	Saham	Mean Return
1	0.003748	11	0.001377
2	0.005254	12	0.000449
3	0.003050	13	0.002775
4	0.002232	14	0.002028
5	0.004411	15	0.002299
6	0.003561	16	0.001821
7	-0.000970	17	0.000457
8	0.001430	18	0.008209
9	0.004703	19	0.004690
10	0.006346	20	-0.000069

Data tersebut dimodelkan dalam portofolio Markowitz kemudian diolah untuk dicari solusi optimumnya menggunakan alat bantu Matlab R2013a dengan algoritma *Cuckoo Search*. Data diproses dengan melakukan *running* sebanyak 20 kali dan solusi yang diambil adalah solusi terbaik (optimum). Berikut ini merupakan algoritma *Cuckoo Search* yang mengacu pada algoritma dasar yang diberikan pada [4].

i). *Input*

d : dimensi/ banyak peubah

n : banyak sarang (individu), biasanya dipakai $10d$

$Pa \in [0,1]$: peluang telur burung *Cuckoo* ditemukan

$N_{itermax}$: iterasi maksimal

eps : batas toleransi kriteria pemberhentian *Chauchy* (*stop criterion*)

Lb : batas bawah

Ub : batas atas

ii). *Proses*

a) Membangkitkan populasi awal sebanyak n sarang (individu), $x_j \in \mathbb{R}^d$ untuk $j = 1, 2, \dots, n$ di daerah definisi.

b) Melakukan evaluasi *fitness*, $Fitness_j = f(x_j)$ untuk $j = 1, 2, \dots, n$.

c) Menetapkan sarang dengan individu yang memiliki nilai *fitness* terbaik dari n sarang yang telah dibangun, yaitu

$$x' = x_{jg} \quad (10)$$

dimana

$$jg = \arg \min_j \text{Fitnes}_j, j = 1, 2, \dots, n$$

dan

$$fmin = f(x_{jg})$$

d) Tetapkan $s = \text{zeros}(d, 1)$ sebagai tempat untuk sarang baru.

e) Tetapkan $g = 1$ sebagai iterasi awal.

f) Tetapkan $r = 1$.

g) *While* $r \neq 0$ dan $g \leq N_itermax$

for $j = 1:n$

- Membangkitkan sarang baru ($s \in \mathbb{R}^d$) dengan *Lévy flight*.

Pilih $\beta = 3/2$ serta definisikan u, v, σ_u, σ_v , dan *step* seperti pada persamaan (7)-(9), sehingga

$$\text{stepsize} = 0.01 * \text{step} * (s - x')$$

$$s = x' + \text{stepsize} * \text{randn}(\text{size}(s))$$

- Memilih satu sarang (individu) dari n sarang (individu) secara acak, katakan x_k .
- Melakukan evaluasi *fitness*, $f = f(s)$ dan penggantian sarang yang tidak berkualitas *if* $f \leq \text{Fitnes}_k$

$$x_k = s$$

$$\text{Fitnes}_k = f$$

end if

- Mengabaikan sarang dengan individu yang tidak berkualitas (jika peluang telur burung *Cuckoo* ditemukan lebih besar dari Pa) dan membangun sarang baru (s) dengan *random walk* untuk menggantikan sarang yang tidak berkualitas.

➤ Bangkitan bilangan acak, $u \in [0, 1]$

Jika $u > Pa$ maka pilih sarang dengan individu yang tidak berkualitas, yaitu yang memiliki nilai *fitness* terburuk.

$$x_l = x_{jg}$$

dimana

$$jg = \arg \max_j \text{Fitnes}_j, j = 1, 2, \dots, n$$

dan

$$fmax = f(x_{jg})$$

Pemilihan x_l seperti ini menjaga agar bukan solusi terbaik yang akan tergantikan $x_l \neq x'$.

➤ Bangun sarang baru (s) dengan *random walk* untuk menggantikan sarang yang tidak berkualitas.

$$s = x_l + 0.05 * \text{randn}(\text{size}(s))$$

➤ Melakukan evaluasi *fitnes* dan penggantian sarang yang tidak berkualitas

$$f = f(s)$$

$$x_l = s$$

$$\text{Fitnes}_l = f$$

- Menentukan sarang dengan individu terbaik sebagai solusi seperti pada (10) sehingga diperoleh x', z' baru dan $fnew$.

end for

h) Menentukan kriteria pemberhentian *Cauchy*.

Kriteria pemberhentian *Cauchy* dilakukan setelah iterasi mencapai setengah dari iterasi maksimum dan dilakukan pengecekan selama 50 iterasi secara berturut-turut.

$$\text{if } g \geq \frac{1}{2}N_{itermax} + 50$$

$$\delta_i = |fnew_i - fnew_{i-1}|, i = g - 49: g$$

dimana g, i adalah indeks iterasi dan $fnew_i$ adalah nilai *fitness* terbaik pada iterasi ke- i . Dalam tesis ini digunakan asumsi bahwa jika selama 50 iterasi secara berturut-turut ($i = g - 49: g$), $\delta_i \leq eps$, maka proses selesai. Sehingga hasil perbandingan δ_i dan eps dikumpulkan sebagai berikut

$$q_i = \text{greater than } (\delta_i, eps), i = g - 49: g$$

Jika $\delta_i > eps$ maka $q_i = 1$ dan jika $\delta_i \leq eps$ maka $q_i = 0$. Kemudian tentukan

$$r = \text{sum}(q_i), i = g - 49: g$$

Jika $r = 0$ maka $\delta_i \leq eps$ untuk $i = g - 49: g$, artinya selama selama 50 iterasi secara berturut-turut $\delta_i \leq eps$ dan proses berhenti. Jika $r \neq 0$ berarti selama 50 iterasi secara berturut-turut masih ada beberapa $\delta_i > eps$ dan proses berlanjut.

end if

i) Melakukan *update* iterasi $g = g + 1$.

j) *end while*

iii). *Output*

Nilai minimum global dari fungsi objektif ($fnew$) dan titik minimum global (sarang terbaik, x').

Sedangkan untuk menyelesaikan masalah optimasi yang melibatkan masalah MINLP, digunakan modifikasi dari Algoritma *Cuckoo Search*, seperti modifikasi yang telah dilakukan pada [8],[9], dan [10]. Modifikasi ini pada dasarnya tidak mengurangi atau menghilangkan sifat dasar dari *Cuckoo Search*. Asumsi dan langkah yang digunakan masih sama dengan asumsi dan langkah dari algoritma *Cuckoo Search*. Kunci utama dalam modifikasi ini adalah dengan melakukan *rounding* terhadap variabel-variabel real, sedemikian sehingga menjadi variabel bulat. Misalkan x adalah variabel real, maka z yang merupakan variabel bulat dari x didefinisikan,

$$z = \text{round}(x)$$

dimana $\text{round}(x)$ adalah bilangan

3. Hasil dan Pembahasan

Algoritma *Cuckoo Search* diterapkan untuk masalah optimasi portofolio (1) dan (2), serta (3) dan (4), dan modifikasi *Cuckoo Search* diterapkan untuk masalah (5) dan (6). Adapun input parameter yang digunakan adalah, $d = 20$, $n = 10d$, $Pa = 0.1$, $N_{itermax} = 200000$, $eps = 10^{-12}$, $rho = 10^2$, dan $mu = 10^5$.

3.1 Masalah Minrisk

Penerapan *Cuckoo Search* untuk masalah Minrisk dengan data 20 saham LQ45, diberikan target *return*, $Rp = 0.01$, syarat batas minimum tiap saham adalah $y_{min} = 0.007$, dan syarat batas maksimum tiap saham adalah $y_{max} = 1$. Setelah dilakukan proses *running* sebanyak 20 kali, didapatkan hasil terbaik sebagai berikut,

TABEL 3. HASIL MINRISK 20 SAHAM LQ45

	Minrisk	Minrisk dengan kendala
--	---------	------------------------

Saham	tanpa <i>short-selling</i> Pers. (1)	<i>buy-in threshold</i>		
		Pers. (3)	Pers. (5)	
			y_i	z_i
1				
2				
4	0.0000	0.0070	0.0070	1
6	0.1061	0.0550	0.0599	1
7	0.1109	0.0845	0.0892	1
8	0.0058	0.0725	0.0723	1
11	0.0000	0.0070	0.0070	1
12	0.0000	0.0070	0.0070	1
13	0.0000	0.0070	0.0070	1
14	0.2037	0.2727	0.2726	1
17	0.1044	0.1676	0.1650	1
20	0.0924	0.0219	0.0214	1
21	0.0646	0.0142	0.0115	1
22	0.0000	0.0070	0.0070	1
23	0.0000	0.0070	0.0070	1
25	0.3121	0.2196	0.2158	1
31	0.0000	0.0070	0.0070	1
32	0.0000	0.0070	0.0070	1
41	0.0000	0.0070	0.0070	1
42	0.0000	0.0150	0.0155	1
V	0.0014	0.0016	0.0016	
Iterasi	31774	60405	139077	
Waktu	551.831564 detik	1072.041277 detik	2254.583281 detik	

Penerapan *Cuckoo Search* untuk masalah Minrisk dengan data 20 saham pertama FTSE, digunakan target *return*, $R_p = 0.004$, syarat batas minimum tiap saham adalah $y_{min} = 0.007$, dan syarat batas maksimum tiap saham adalah $y_{max} = 1$. Setelah dilakukan proses *running* sebanyak 20 kali, didapatkan hasil terbaik sebagai berikut,

TABEL 4. HASIL MINRISK 20 SAHAM PERTAMA FTSE

Saham	Minrisk tanpa <i>short-selling</i> Pers. (1)	Minrisk dengan kendala <i>buy-in threshold</i>		
		Pers. (3)	Pers. (5)	
			y_i	z_i
1				
2				
3	0.0275	0.0122	0.0122	1
4	0.2594	0.2262	0.2264	1
5	0.1127	0.1218	0.1218	1
6	0.0000	0.0099	0.0099	1
7	0.0018	0.0118	0.0118	1
8	0.0000	0.0070	0.0070	1
9	0.0000	0.0090	0.0090	1
10	0.0000	0.0070	0.0070	1
11	0.1084	0.0909	0.0909	1
12	0.0653	0.0825	0.0825	1
13	0.0098	0.0189	0.0189	1
14	0.0189	0.0130	0.0129	1
15	0.0345	0.0129	0.0129	1
16	0.0272	0.0605	0.0604	1
17	0.1133	0.0995	0.0995	1
18	0.0001	0.0105	0.0105	1
19	0.0138	0.0327	0.0327	1
20	0.1005	0.1030	0.1031	1
21	0.0000	0.0106	0.0106	1
22	0.1067	0.0606	0.0606	1

20			
V	3.1918e-004	3.28484e-04	3.8990e-04
Iterasi	58274	72996	113222
Waktu	755.002004 detik	1138.368380 detik	2039.097756 detik

3.2 Masalah Maxret

Penerapan *Cuckoo Search* untuk masalah Maxret dengan data 20 saham LQ45, diberikan besar risiko yang berani diterima, $Va = 0.0025$, syarat batas minimum tiap saham adalah $y_{min} = 0.007$, dan syarat batas maksimum tiap saham adalah $y_{max} = 1$. Setelah dilakukan proses *running* sebanyak 20 kali, didapatkan hasil terbaik sebagai berikut,

TABEL 5. HASIL MAXRET 20 SAHAM LQ45

Saham	Maxret tanpa short-selling Pers. (2)	Maxret dengan kendala buy-in threshold		
		Pers. (4)	Pers. (6)	
			y_i	z_i
1				
2	0.0000	0.0070	0.0070	1
4	0.1430	0.0773	0.0753	1
6	0.0000	0.0416	0.0412	1
7	0.0000	0.0426	0.0433	1
8	0.0000	0.0070	0.0070	1
11	0.0000	0.0070	0.0070	1
12	0.0000	0.0070	0.0070	1
13	0.0000	0.0070	0.0070	1
14	0.4368	0.2110	0.2117	1
	0.2873	0.4136	0.4142	1
17	0.0000	0.0136	0.0138	1
20	0.0000	0.0101	0.0100	1
21	0.0000	0.0070	0.0070	1
22	0.0000	0.0070	0.0070	1
23	0.0000	0.0070	0.0070	1
25	0.1330	0.1063	0.1066	1
31	0.0000	0.0070	0.0070	1
32	0.0000	0.0070	0.0070	1
41	0.0000	0.0070	0.0070	1
42	0.0000	0.0070	0.0070	1
R	0.0111	0.0104	0.0104	
Iterasi	58040	96562	131552	
Waktu	753.276862 detik	1635.144604 detik	2108.078123 detik	

Penerapan *Cuckoo Search* untuk masalah Maxret dengan data 20 saham pertama FTSE, diberikan besar risiko yang berani diterima, $Va = 0.0006$, syarat batas minimum tiap saham adalah $y_{min} = 0.007$, dan syarat batas maksimum tiap saham adalah $y_{max} = 1$. Setelah dilakukan proses *running* sebanyak 20 kali, didapatkan hasil terbaik sebagai berikut,

TABEL 6. HASIL MAXRET 20 SAHAM PERTAMA FTSE

Saham	Maxret tanpa short-selling	Maxret dengan kendala buy-in threshold
-------	----------------------------	--

	Pers. (2)	Pers. (4)	Pers. (6)	
			y_i	z_i
1		0.0070		
2		0.1349		
3	0.0000	0.0070	0.0070	1
4	0.2936	0.0070	0.1351	1
5	0.0000	0.0180	0.0070	1
6	0.0000	0.0079	0.0070	1
7	0.0000	0.0000	0.0180	1
8	0.0000	0.0070	0.0079	1
9	0.0000	0.1357	0.0070	0
10	0.0677	0.1551	0.1358	1
11	0.2050	0.0070	0.1549	1
12	0.0000	0.0070	0.0070	1
13	0.0000	0.0070	0.0070	1
14	0.0000	0.0070	0.0070	1
15	0.0000	0.0070	0.0070	1
16	0.0000	0.0070	0.0070	1
17	0.0000	0.0070	0.0070	1
18	0.4337	0.0070	0.4482	1
19	0.0000	0.4482	0.0161	1
20	0.0000	0.0161	0.0070	1
		0.0070		
<i>R</i>	0.0067	0.0064	0.0063	
Iterasi	34211	133585	105422	
Waktu	455.867597 detik	2326.558463 detik	1966.575885 detik	

Dari hasil optimasi portofolio pada Tabel 3 sampai Tabel 6, terlihat jelas bahwa pada optimasi portofolio tanpa kendala, yaitu masalah Minrisk dan Maxret tanpa *short-selling*, masih dijumpai proporsi saham yang sangat kecil ($y_i < y_{min}$). Sedangkan pada optimasi portofolio dengan kendala *buy-in threshold* sudah tidak dijumpai proporsi saham yang lebih kecil dari y_{min} . Proporsi saham tersebut menjadi lebih besar atau sama dengan y_{min} , atau sama dengan nol. Serta dapat diamati pula bahwa hasil optimasi portofolio dengan kendala *buy-in threshold* yang dikerjakan dengan metode *Cuckoo Search* dan modifikasi *Cuckoo Search* memberikan hasil yang dekat atau bisa dikatakan sama.

4. Simpulan dan Saran

4.1 Simpulan

Cuckoo Search merupakan metode metaheuristik yang baik untuk menyelesaikan masalah optimasi portofolio dengan kendala *buy-in threshold*. Pada dasarnya *Cuckoo Search* hanya digunakan untuk menyelesaikan masalah optimasi portofolio yang melibatkan variabel kontinu. Pada penelitian ini, telah dilakukan modifikasi terhadap *Cuckoo Search* sehingga dapat digunakan untuk menyelesaikan masalah optimasi yang melibatkan variabel bilangan bulat (*Mixed Integer Nonlinear Programming*). Masalah optimasi portofolio dengan kendala *buy-in threshold* didefinisikan dalam dua jenis masalah, yaitu masalah optimasi portofolio yang hanya melibatkan variabel-variabel kontinu, dan masalah optimasi portofolio yang melibatkan variabel-variabel bulat (MINLP). Perhitungan kedua masalah tersebut dengan menggunakan *Cuckoo Search* dan modifikasi *Cuckoo Search* secara berturut-turut, memberikan hasil yang sama. Hal ini membuktikan bahwa *Cuckoo Search* selain dapat digunakan untuk menyelesaikan masalah optimasi portofolio yang melibatkan variabel-variabel kontinu, modifikasi dari *Cuckoo Search* juga dapat digunakan untuk menyelesaikan masalah optimasi portofolio yang melibatkan variabel-variabel bulat (MINLP).

4.2 Saran

Berdasarkan hasil penelitian, *Cuckoo Search* merupakan metode metaheuristik yang baik untuk menyelesaikan masalah optimasi portofolio, khususnya masalah optimasi portofolio dengan kendala *buy-in threshold*. Selanjutnya, *Cuckoo Search* dapat pula digunakan untuk menyelesaikan masalah optimasi portofolio dengan berbagai kendala yang lain, seperti kendala *cardinality* dan *roundlot*.

5. Daftar Pustaka

- [1] Markowitz, H. 1952. Portofolio Selection. *The Journal of Finance*, Vol. 7, pp. 77-91.
- [2] Biggs, M. C. Bartholomew dan Kane, S. J. 2007. A Global Optimization Problem in Portofolio Selection. Springer. *Management Science* (2009) 6:329-345 DOI 10.1007/s10287-006-0038-4.
- [3] Jobst, N.J., Horniman, M.D., Lucas, C.A., dan Mitra, G. 2001. Computational Aspects of Alternative Portfolio Selection Models in The Presence of Discrete Asset Choice Constraints. *Quantitative Finance* 1, pp. 1-13.
- [4] Yang, X. S. dan Deb, S. 2009. Cuckoo Search via Lévy Flights. *Proceeding of World Congres on Nature and Biologically Inspired Computing (NaBIC 2009, India)*, IEEE Publications, USA, pp. 210-214.
- [5] http://www.idx.co.id/Portals/0/StaticData/MarketInformation/ListOfSecurities/IndexConstituent/LQ45/20140124_Feb-Juli-2014.pdf, diakses pada tanggal 8 Mei 2017, pukul 14.00 WIB.
- [6] <http://finance.yahoo.com/>, diakses pada tanggal 8 Mei 2014, pukul 13.30 WIB.
- [7] <http://people.brunel.ac.uk/~mastjjb/jeb/orlib/files/>, diakses pada tanggal 8 Mei 2017, pukul 13.00 WIB.
- [8] Muzdalifah, Lilik. 2016. Binary Cuckoo Search untuk Optimasi Portofolio dengan Kendala Cardinality. *Jurnal AdMathEdu*, Vol.6, No. 1, Hal. 25-32.
- [9] Hongqing Zheng, Yongquan Zhou , Sucai He, Xinxin Ouyang. 2012. A Discrete Cuckoo Search Algorithm for Solving Knapsack Problems. *Advances in information Sciences and Service Sciences(AISS)*, Vol. 4.
- [10] Yanhong Feng, Ke Jia, dan Yichao He. 2014. An Improved Hybrid Encoding Cuckoo Search Algorithm for 0-1 Knapsack Problems. *The Journal of Computational Intelligence and Neuroscience*, Vol. 2014, Article ID 970456, 9 pages.