

PENERAPAN ALGORITMA GREEDY FRACTIONAL KNAPSACK PADA PEMILIHAN TANAH LIAT UNTUK PRODUKSI GERABAH DI BANTUL

Erfannindya Ade Noventyas^{1*}, Gunawan Wijayanto²,
M Rizky Ramadhan³, M Yusuf⁴, Tedy Setiadi⁵

¹²³⁴⁵Universitas Ahmad Dahlan Yogyakarta

*2300018279@webmail.uad.ac.id

Abstract

The pottery industry in Bantul faces challenges in determining a cost-efficient combination of clay raw materials, considering production capacity limitations as well as price and weight variations for each type of clay. This study aims to identify the most economical clay composition for pottery production using the Greedy Fractional Knapsack algorithm. The method employed is a quantitative approach based on algorithmic simulation involving five types of clay, with a total capacity limit of 100 kilograms. Four sorting strategies were applied in the experiment: arranging by weight from lightest to heaviest, from heaviest to lightest, by price from cheapest to most expensive, and by the price-to-weight ratio (density). The results indicate that the density-based strategy yields the most optimal outcome, with a total cost of IDR 48,000. The optimal composition consists of 100% red clay, black clay, and gray clay, along with 33% brown clay, while white clay was eliminated due to inefficiency. These findings suggest that the Greedy Fractional Knapsack algorithm is effective as a decision-support tool for raw material logistics in the handicraft industry.

Keywords: *Greedy, Knapsack, Logistics, Pottery, Clay*

Abstrak

Industri gerabah di Bantul menghadapi tantangan dalam menentukan kombinasi bahan baku tanah liat yang efisien secara biaya, mengingat keterbatasan kapasitas produksi dan variasi harga serta berat tiap jenis tanah liat. Penelitian ini bertujuan untuk mengidentifikasi komposisi tanah liat yang paling ekonomis untuk produksi gerabah dengan menggunakan algoritma *greedy Fractional Knapsack*. Metode yang digunakan adalah pendekatan kuantitatif berbasis simulasi algoritmik terhadap lima jenis tanah liat, dengan pembatasan total kapasitas sebesar 100 kilogram. Empat strategi pengurutan diterapkan dalam eksperimen: berdasarkan berat dari ringan ke berat, dari berat ke ringan, harga dari murah ke mahal, serta rasio harga terhadap berat (*density*). Hasil penelitian menunjukkan bahwa strategi berdasarkan *density* memberikan hasil paling optimal, dengan total biaya sebesar Rp48.000. Komposisi optimal terdiri dari 100% tanah liat merah, hitam, dan abu, serta 33% tanah liat coklat, sedangkan tanah liat putih dieliminasi karena tidak efisien. Temuan ini mengindikasikan bahwa algoritma *greedy Fractional Knapsack* efektif digunakan sebagai alat bantu pengambilan keputusan logistik bahan baku dalam industri kerajinan.

Kata Kunci: *Greedy, Knapsack, Logistik, Gerabah, Tanah Liat*

Pendahuluan

Industri kerajinan gerabah di Bantul memainkan peran krusial dalam mendukung perekonomian lokal sebagai salah satu sektor ekonomi kreatif. Proses pembuatan gerabah memerlukan bahan baku utama, yaitu tanah liat, yang tersedia dalam berbagai jenis dengan karakteristik yang berbeda, termasuk berat dan harga. Keberagaman ini menciptakan tantangan tersendiri dalam menentukan kombinasi bahan yang paling efisien dari segi biaya, terutama karena kapasitas produksi memiliki batasan tertentu. Oleh karena itu, penting untuk menerapkan metode seleksi bahan baku yang tidak hanya memperhatikan kualitas, tetapi juga aspek ekonomis, demi memastikan keberlangsungan dan daya saing industri ini.

Salah satu metode yang efektif untuk mengatasi masalah optimasi ini adalah algoritma *greedy Fractional Knapsack*. Algoritma ini dirancang untuk menyelesaikan masalah knapsack dalam bentuk fraksional, di mana item, seperti tanah liat, dapat diambil dalam proporsi tertentu sesuai dengan kapasitas yang ada. Berbeda dengan algoritma *Integer Knapsack*, pendekatan fraksional ini memberikan lebih banyak fleksibilitas dalam pengambilan keputusan, yang sangat bermanfaat dalam konteks logistik bahan baku (Ali, 2017).

Dalam bidang informatika terapan, algoritma *greedy* telah diterapkan secara luas untuk mengatasi berbagai masalah praktis, seperti pengangkutan barang (Ambarwari & Yanti, 2016), pemilihan bahan yang optimal (Devita & Wibawa, 2020), serta sistem pendukung keputusan dalam logistik (Rianti et al., 2020). Salah satu keunggulan utama dari algoritma ini adalah kemampuannya untuk memberikan solusi yang hampir optimal dengan waktu pemrosesan yang efisien. Selain itu, kemudahan implementasinya dalam bahasa pemrograman seperti Python menjadikannya pilihan yang ideal bagi industri kecil dan menengah yang memerlukan sistem otomatisasi berbasis teknologi informasi.

Industri gerabah di Bantul sedang menghadapi tantangan dalam mempertahankan efisiensi produksi, terutama karena keterbatasan sumber daya seperti biaya bahan baku dan kapasitas muatan. Oleh karena itu, pemilihan tanah liat yang optimal menjadi faktor krusial yang dapat mempengaruhi biaya produksi dan kelangsungan usaha secara signifikan. Dalam situasi ini, para pelaku industri perlu mengadopsi pendekatan yang berbasis data dan teknologi untuk secara objektif dan sistematis meningkatkan efisiensi operasional mereka (Setiawati et al., 2024).

Masalah optimasi yang berkaitan dengan pemilihan barang dengan batasan kapasitas dikenal dalam literatur sebagai masalah *knapsack*. Salah satu variasi dari masalah ini, yaitu *Fractional Knapsack*, memungkinkan pemilihan barang dalam bentuk pecahan, sehingga sangat sesuai untuk situasi di mana bahan dapat dibagi, seperti tanah liat. Pendekatan algoritmik ini telah terbukti efektif dalam menyelesaikan masalah dunia nyata, seperti pemilihan barang logistik (Nasution & Ahyaningsih, 2024) dan penentuan prioritas distribusi (Herryansyah, 2024).

Pendekatan greedy sering dipilih sebagai metode untuk menyelesaikan masalah optimasi karena kemudahan dalam penerapannya dan kecepatan dalam menghasilkan solusi yang memuaskan. Penelitian sebelumnya menunjukkan bahwa algoritma *greedy* efektif dalam memaksimalkan utilitas di berbagai sektor, termasuk transportasi barang, pemilihan produk, dan manajemen distribusi. Hal ini terutama berlaku ketika algoritma ini dipadukan dengan bahasa pemrograman Python, yang dikenal fleksibel dan mudah diakses oleh pengguna baik yang memiliki latar belakang teknis maupun non-teknis (Ilham & Saputra, 2023; Piliang, 2017)

Penelitian ini sangat penting untuk menawarkan solusi berbasis algoritma yang dapat langsung diterapkan oleh pelaku usaha kecil dan menengah (UKM). Dengan menggunakan pendekatan eksperimental yang melibatkan data karakteristik dari lima jenis tanah liat yang umum di Bantul, studi ini bertujuan untuk menjawab pertanyaan: bagaimana cara menentukan kombinasi tanah liat yang paling ekonomis sambil tetap memenuhi batas berat maksimum produksi? Diharapkan, jawaban atas pertanyaan ini akan memberikan solusi praktis yang dapat diimplementasikan.

Metode Penelitian

Studi ini mengadopsi pendekatan kuantitatif melalui metode eksperimen yang berfokus pada simulasi algoritma untuk mengatasi tantangan dalam mengoptimalkan pemilihan kombinasi bahan baku tanah liat. Pendekatan ini diterapkan dalam industri kerajinan gerabah di Bantul, yang menghadapi batasan kapasitas bahan maksimum serta variasi harga dan berat dari bahan baku yang ada (Hikmawati, 2020; Ismayani, 2019).

1. Data dan Variabel

Penelitian ini memanfaatkan data sekunder yang disusun dalam format data Python. Data tersebut diperoleh dari studi dokumentasi dan referensi teknis terkait karakteristik lima jenis tanah liat yang umum digunakan dalam produksi gerabah di Bantul. Setiap jenis tanah liat dievaluasi berdasarkan tiga kriteria utama:

- Nama tanah liat, yang berfungsi untuk mengidentifikasi jenis bahan.
- Berat tanah liat (w_i), yang diukur dalam kilogram.
- Harga atau biaya (p_i), yang dinyatakan dalam rupiah (Rp).

Struktur data yang digunakan ditunjukkan pada Gambar 1, yang memuat daftar jenis tanah liat dalam format Python untuk mendukung proses analisis.

```
data_tanah_liat = [  
    (30, 15, "Tanah Liat Merah"),  
    (20, 30, "Tanah Liat Putih"),  
    (40, 20, "Tanah Liat Hitam"),  
    (15, 9, "Tanah Liat Cokelat"),  
    (25, 10, "Tanah Liat Abu")  
]
```

Gambar 1. Data Tanah Liat

2. Teknik Analisis Data

Analisis data dilakukan dengan menggunakan algoritma *greedy Fractional Knapsack*, yang memungkinkan pemilihan item dalam bentuk fraksional untuk memaksimalkan total kapasitas (dalam hal ini 100 kg) dengan biaya yang serendah mungkin. Dalam eksperimen ini, empat strategi pengurutan diterapkan:

- (1) Mengurutkan berdasarkan berat dari yang teringan hingga terberat.
- (2) Mengurutkan berdasarkan berat dari yang terberat hingga teringan.
- (3) Mengurutkan berdasarkan harga dari yang termurah hingga termahal.
- (4) Mengurutkan berdasarkan rasio harga terhadap berat (p_i/w_i), yang dikenal sebagai density.

Setiap strategi memungkinkan algoritma untuk menentukan kombinasi tanah liat yang dapat diambil secara utuh atau dalam bentuk pecahan, hingga mencapai batas maksimum kapasitas 100 kg.

Hasil dan Pembahasan

A. Algoritma Dan Perancangan Program

Algoritma *greedy* untuk Knapsack Fraksional dirancang agar memungkinkan pemilihan item dalam bentuk fraksional. Implementasi algoritma ini dilakukan dengan pendekatan pemrograman Python, dan memiliki fungsi utama sebagai berikut ditunjukkan pada gambar 2.

```
def format_rupiah(angka):  
    return f"Rp {int(angka):,}".replace(",", ".")  
  
# Fungsi utama algoritma greedy Fractional Knapsack  
def knapsack_greedy(data_tanah, berat_maksimal, fungsi_sortir,  
   urut_mundur=False, bisa_fractions=True):  
    data_terurut = sorted(data_tanah, key=fungsi_sortir,  
    reverse=urut_mundur)
```

```
total_berat = 0
total_biaya = 0
tanah_terpilih = []

for tanah in data_terurut:
    berat, harga, nama = tanah
    if total_berat + berat <= berat_maksimal:
        tanah_terpilih.append((nama, berat, harga, 1.0))
        total_berat += berat
        total_biaya += harga
    elif bisa_fractions:
        sisa_berat = berat_maksimal - total_berat
        if sisa_berat <= 0:
            break
        fraksi_diambil = sisa_berat / berat
        biaya_fraksi = harga * fraksi_diambil
        tanah_terpilih.append((nama, sisa_berat, biaya_fraksi,
round(fraksi_diambil, 2)))
        total_berat += sisa_berat
        total_biaya += biaya_fraksi
        break

return tanah_terpilih, total_berat, total_biaya

# Fungsi cetak tabel perbandingan strategi
def cetak_tabel_komparasi(data_awal, hasil1, hasil2, hasil3, hasil4,
hasil_optimal):
    print(f"{'No':<3} {'Nama Tanah':<20} {'Berat':<7} {'Harga':<12}
{'p/w':<7} "
        f"{'Ringan':<10} {'Berat':<10} {'Murah':<10} {'Density':<12}
{'Optimal':<10}")
    print("-" * 120)
```

```
def cari_persen(nama, hasil):
    for item in hasil:
        if item[0] == nama:
            return item[3]
    return 0.0

for i, tanah in enumerate(data_awal, start=1):
    wi, pi, nama = tanah
    density = round(pi / wi, 2)
    gr1 = cari_persen(nama, hasil1)
    gr2 = cari_persen(nama, hasil2)
    gr3 = cari_persen(nama, hasil3)
    gr4 = cari_persen(nama, hasil4)
    opt = cari_persen(nama, hasil_optimal)

    print(f"{i:<3} {nama:<20} {wi:<7} {format_rupiah(pi * 1000):<12}
{density:<7} "
          f"{gr1:<10} {gr2:<10} {gr3:<10} {gr4:<12} {opt:<10}")

print("-" * 120)

def get_total(data):
    berat = sum(d[1] for d in data)
    biaya = sum(d[2] for d in data) * 1000 # ke rupiah
    return round(berat, 2), round(biaya)

b1, p1 = get_total(hasil1)
b2, p2 = get_total(hasil2)
b3, p3 = get_total(hasil3)
b4, p4 = get_total(hasil4)
bopt, popt = get_total(hasil_optimal)

# Cetak total berat
```

```
print(f"{'Total Berat (Kg)':<53} {b1:<10} {b2:<10} {b3:<10} {b4:<12}
{bopt:<10}")

# Cetak total biaya
print(f"{'Total Biaya':<53} {format_rupiah(p1):<10}
{format_rupiah(p2):<10} "
      f"{format_rupiah(p3):<10} {format_rupiah(p4):<12}
{format_rupiah(popt):<10}")

# Data tanah liat
data_tanah_liat = [
    (30, 15, "Tanah Liat Merah"),
    (20, 30, "Tanah Liat Putih"),
    (40, 20, "Tanah Liat Hitam"),
    (15, 9, "Tanah Liat Cokelat"),
    (25, 10, "Tanah Liat Abu")
]
maksimal_berat = 100
# Strategi greedy
hasil_gr1, _, _ = knapsack_greedy(data_tanah_liat, maksimal_berat,
fungsi_sortir=lambda x: x[0], urut_mundur=False)
hasil_gr2, _, _ = knapsack_greedy(data_tanah_liat, maksimal_berat,
fungsi_sortir=lambda x: x[0], urut_mundur=True)
hasil_gr3, _, _ = knapsack_greedy(data_tanah_liat, maksimal_berat,
fungsi_sortir=lambda x: x[1], urut_mundur=False)
hasil_gr4, _, _ = knapsack_greedy(data_tanah_liat, maksimal_berat,
fungsi_sortir=lambda x: x[1] / x[0], urut_mundur=False)

# Tentukan solusi optimal (paling hemat biaya)
semua_biaya = {
    "Ringan": sum(d[2] for d in hasil_gr1),
    "Berat": sum(d[2] for d in hasil_gr2),
    "Murah": sum(d[2] for d in hasil_gr3),
```

```
"Density": sum(d[2] for d in hasil_gr4)
}
metode_opt = min(semua_biaya, key=semua_biaya.get)

hasil_optimal = {
    "Ringan": hasil_gr1,
    "Berat": hasil_gr2,
    "Murah": hasil_gr3,
    "Density": hasil_gr4
}[metode_opt]

# Tampilkan hasil tabel akhir
print("\n=== PERBANDINGAN STRATEGI GREEDY FRACTIONAL KNAPSACK ===")
cetak_tabel_komparasi(
    data_tanah_liat,
    hasil_gr1,
    hasil_gr2,
    hasil_gr3,
    hasil_gr4,
    hasil_optimal
)

# Tampilkan strategi paling hemat
print("\n=== SOLUSI OPTIMAL ===")
total_berat_opt = round(sum(d[1] for d in hasil_optimal), 2)
total_biaya_opt = round(sum(d[2] for d in hasil_optimal) * 1000) # Dalam
rupiah
print(f"Metode paling hemat biaya: {metode_opt}")
print(f"Total berat optimal      : {total_berat_opt} kg")
print(f"Total biaya optimal       : {format_rupiah(total_biaya_opt)}")
```

Gambar 2. Kode Python *Algoritma Greedy* Fraksional Knapsack

Fungsi `knapsack_greedy` yang disebutkan pada gambar 3 dibawah memungkinkan pemilihan item (tanah liat) secara utuh atau sebagian, tergantung pada kapasitas berat_maksimal yang tersisa. Proses pengurutan awal pada `data_terurut` mencerminkan penerapan strategi greedy yang digunakan.

```
def knapsack_greedy(data_tanah, berat_maksimal, fungsi_sortir,  
urut_mundur=False, bisa_fractions=True):
```

Gambar 3. Fungsi Utama Algoritma *Greedy Fractional Knapsack*

B. Strategi Eksperimen

Studi ini mengimplementasikan empat strategi heuristik yang berlandaskan algoritma *greedy Fractional Knapsack* untuk menilai efisiensi dalam pemilihan kombinasi bahan tanah liat sesuai dengan kriteria pengurutan tertentu. Setiap strategi dirancang untuk mengeksplorasi dampak metode pengurutan terhadap total biaya, dengan mempertimbangkan batas maksimum kapasitas berat sebesar 100 kilogram. Empat strategi greedy diterapkan terhadap data yang sama dengan urutan pengurutan berbeda:

1. Strategi 1 Pengurutan Berdasarkan (Ringan Ke Berat)

Dalam strategi ini, data tanah liat diurutkan dari berat teringan hingga terberat. Tujuannya adalah untuk mengisi kapasitas dengan sebanyak mungkin jenis bahan berukuran kecil terlebih dahulu.

Kode implementasi pada gambar 4 :

```
hasil_gr1, _, _ = knapsack_greedy(data_tanah_liat, maksimal_berat,  
fungsi_sortir=lambda x: x[0], urut_mundur=False)
```

Gambar 4. Strategi *Greedy* Ringan ke Berat

2. Strategi 2 Pengurutan Berdasarkan (Berat Ke Ringan)

Sebaliknya, strategi ini memprioritaskan jenis tanah liat dengan berat terbesar terlebih dahulu, dengan asumsi bahwa bahan berat lebih efisien dalam memenuhi kapasitas lebih cepat.

Kode implementasi pada gambar 5 :

```
hasil_gr2, _, _ = knapsack_greedy(data_tanah_liat, maksimal_berat,  
fungsi_sortir=lambda x: x[0], urut_mundur=True)
```

Gambar 5. Strategi *Greedy* Berat ke Ringan

3. Strategi 3 Pengurutan Berdasarkan Harga (Termurah ke Termahal)

Strategi ini menyusun data berdasarkan harga dari yang paling murah hingga yang paling mahal. Tujuan pendekatan ini adalah untuk mengutamakan bahan dengan biaya rendah terlebih dahulu.

Kode implementasi pada gambar 6 :

```
hasil_gr3, _, _ = knapsack_greedy(data_tanah_liat, maksimal_berat,  
fungsi_sortir=lambda x: x[1],urut_mundur=False)
```

Gambar 6. Strategi *Greedy* Termurah ke Termahal

4. Strategi 4 pengurutan berdasarkan rasio harga terhadap berat. harga per kg termurah (π/w_i)

Strategi ini mengurutkan bahan berdasarkan efisiensi biaya per kilogram (rasio harga terhadap berat), dari yang paling rendah ke yang paling tinggi. Pendekatan ini dianggap paling optimal dalam konteks efisiensi biaya.

Kode implementasi pada gambar 7 :

```
hasil_gr4, _, _ = knapsack_greedy(data_tanah_liat, maksimal_berat,  
fungsi_sortir=lambda x: x[1] / x[0],urut_mundur=False)
```

Gambar 7. Strategi *Greedy* Harga per Kg Termurah

Setelah menerapkan keempat strategi, analisis dilakukan terhadap hasil masing-masing berdasarkan total biaya yang dikeluarkan. Strategi dengan total biaya terendah diidentifikasi sebagai solusi paling optimal untuk masalah pemilihan bahan baku tanah liat dalam industri gerabah. Solusi ini selanjutnya menjadi landasan untuk pengambilan keputusan logistik yang lebih efisien.

Algoritma *greedy* untuk *Fraksional Knapsack* digunakan untuk mengatasi masalah optimasi dalam pemilihan bahan baku tanah liat yang bervariasi dalam jenis, berat, dan harga. Jika kapasitas yang tersedia masih mencukupi, bahan baku dapat diambil secara utuh atau sebagian, tergantung pada sisa kapasitas yang ada. Karakteristik *fraksional* dari algoritma ini memungkinkan pengambilan sebagian dari satu jenis tanah liat. Pada gambar 8 terdapat cuplikan kode yang menggambarkan elemen-elemen penting dalam proses seleksi berdasarkan kapasitas yang tersisa:

```

elif bisa_fractions:
    sisa_berat = berat_maksimal - total_berat
    if sisa_berat <= 0:
        break
    fraksi_diambil = sisa_berat / berat
    biaya_fraksi = harga * fraksi_diambil
    tanah_terpilih.append((nama, sisa_berat, biaya_fraksi,
round(fraksi_diambil, 2)))

```

Gambar 8. Potongan Kode *Fractional Knapsack*

Tabel yang menyajikan perbandingan strategi heuristik memperlihatkan hasil simulasi penerapan algoritma *greedy Fractional Knapsack*. Setiap strategi memiliki kriteria prioritas dalam pemilihan, antara lain:

- a. Ringan: mengurutkan berdasarkan berat terendah terlebih dahulu.
- b. Berat: mengurutkan berdasarkan berat tertinggi terlebih dahulu.
- c. Murah: mengurutkan berdasarkan harga terendah terlebih dahulu.
- d. Density: mengurutkan berdasarkan rasio biaya per kilogram, yaitu harga dibagi berat (pi/wi).

Tabel 1 berikut ini menunjukkan persentase setiap jenis tanah liat yang dipilih untuk masing-masing strategi, serta total berat dan total biaya yang terkait.

=== PERBANDINGAN STRATEGI <i>GREEDY FRACTIONAL KNAPSACK</i> ===									
N o	Nama Tanah	Berat	Harga	p/w	Ringan	Berat	Murah	Densit y	Optim al
1	Tanah Liat Merah	30	Rp 15.000	0.5	1.0	1.0	1.0	1.0	1.0
2	Tanah Liat Putih	20	Rp 30.000	1.5	1.0	0.25	0.0	0.0	0.0
3	Tanah Liat Hitam	40	Rp 20.000	0.5	0.25	1.0	0.75	1.0	1.0
4	Tanah Liat Cokelat	15	Rp 9.000	0.6	1.0	0.0	1.0	0.33	0.33
5	Tanah Liat Abu	25	Rp 10.000	0.4	1.0	1.0	1.0	1.0	1.0

Total Berat (kg)	100	100	100	100	100
Total Biaya	Rp 69.000	Rp 52.500	Rp 49.000	Rp 48.000	Rp 48.000
<p>=== SOLUSI OPTIMAL ===</p> <p>Metode paling hemat biaya : Density</p> <p>Total berat optimal : 100 kg</p> <p>Total biaya optimal : Rp 48.000</p>					

Tabel 1. Hasil *Output* Program

Hasilnya menunjukkan bahwa strategi yang mengutamakan urutan berdasarkan rasio harga per berat (*density*) adalah yang paling efisien. Strategi ini mampu menampung maksimum 100 kg dengan biaya terendah sebesar Rp 48.000. Untuk mencapai efisiensi maksimal, komposisi optimal terdiri dari 100% Tanah Liat Merah, Hitam, dan Abu, serta 33% dari Tanah Liat Cokelat. Sementara itu, Tanah Liat Putih tidak dimanfaatkan karena memiliki rasio biaya terhadap berat yang tertinggi, yaitu 1.5.

Kesimpulan

Penelitian ini membahas penerapan algoritma *greedy Fractional Knapsack* dalam menentukan kombinasi bahan baku tanah liat yang paling efisien untuk produksi gerabah di Bantul. Melalui pendekatan eksperimental berbasis simulasi, algoritma diuji menggunakan empat strategi heuristik pengurutan data yang berbeda, yaitu berdasarkan berat (ringan ke berat), (berat ke ringan), harga (murah ke mahal), serta rasio harga terhadap berat (*Density*). Masing-masing strategi menunjukkan hasil komposisi bahan yang berbeda, baik dari segi proporsi jenis tanah liat yang digunakan maupun total biaya yang dihasilkan.

Dari hasil simulasi, strategi berbasis *density* terbukti menghasilkan biaya paling rendah, yaitu sebesar Rp 48.000 untuk kapasitas maksimum 100 kilogram. Komposisi optimal mencakup 100% Tanah Liat Merah, Hitam, dan Abu, serta 33% dari Tanah Liat Cokelat. Sementara itu, Tanah Liat Putih tidak digunakan sama sekali karena memiliki rasio harga per kilogram yang paling tinggi, sehingga tidak efisien secara ekonomis. Temuan ini memperlihatkan bahwa strategi pengurutan berdasarkan efisiensi biaya per satuan berat memberikan dampak yang signifikan terhadap penghematan anggaran bahan baku.

Implikasi praktis dari penelitian ini sangat penting bagi para pelaku industri kerajinan, terutama dalam hal pengambilan keputusan terkait logistik. Penggunaan algoritma *greedy Fractional Knapsack* dapat berfungsi sebagai alat bantu yang efektif dalam merencanakan pembelian bahan baku secara terstruktur dan terukur, terutama ketika dihadapkan pada batasan kapasitas dan anggaran. Dengan memanfaatkan algoritma ini, produsen dapat menemukan kombinasi bahan yang tidak hanya memenuhi target produksi, tetapi juga secara optimal mengurangi biaya operasional.

Oleh karena itu, dapat disimpulkan bahwa algoritma *greedy Fractional Knapsack* merupakan pendekatan yang efisien dalam mengatasi masalah optimasi bahan baku dalam konteks produksi terbatas, serta dapat memberikan keuntungan strategis dalam meningkatkan efisiensi biaya di sektor industri kerajinan, seperti dalam produksi gerabah di Bantul.

Daftar Pustaka

- Ali, N. A. (2017). *Penerapan Algoritma Genetika dan Perbandingannya Dengan Algoritma greedy dalam Peneyelesaian Knapsack Problem Skripsi*. Skripsi. Fakultas Sains dan Teknologi. Universitas Islam Negeri Alauddin <https://core.ac.uk/download/pdf/198220238.pdf>
- Ambarwari, A., & Yanti, N. W. (2016). *Penerapan Algoritma greedy Pada Permasalahan Knapsack Untuk Optimasi Pengangkutan Peti Kemas*. https://www.researchgate.net/profile/Agus-Ambarwari/publication/290797775_Penerapan_Algoritma_Greedy_Pada_Permasalahan_Knapsack_Untuk_Optimasi_Pengangkutan_Peti_Kemas/links/569b9fb108ae748dfb1020ab/Penerapan-Algoritma-Greedy-Pada-Permasalahan-Knapsack-Untuk-Optimasi-Pengangkutan-Peti-Kemas.pdf
- Devita, R. N., & Wibawa, A. P. (2020). Teknik Teknik Optimasi Knapsack Problem. *Sains, Aplikasi, Komputasi Dan Teknologi Informasi*, 2(1), 35–40. <http://dx.doi.org/10.30872/jsakti.v2i1.3299>
- Herryansyah, H. (2024). Sistem Penunjang Keputusan Pemilihan Laba Optimal Dalam Keterbatasan Anggaran Melalui Algoritma Knapsack Dengan Python. *Speed-Sentra Penelitian Engineering Dan Edukasi*, 16(3). <http://speed.web.id/jurnal/index.php/speed/article/view/824>
- Hikmawati, F. (2020). *Metodologi penelitian*. Rajawali Press. <https://digilib.uinsgd.ac.id/31676/1/Metodologi%20Penelitian.pdf>
- Ilham, M. F. N., & Saputra, A. (2023). Seleksi Penerimaan Mahasiswa Baru Dengan Metode Pemecahan Masalah *Algoritma greedy* Menggunakan Python. *Jurnal Rekayasa Teknologi Informasi (JURTI)*, 7(1), 32–38. <http://dx.doi.org/10.30872/jurti.v7i1.9566>
- Ismayani, A. (2019). *Metodologi penelitian*. Syiah Kuala University Press. <https://books.google.com/books?hl=en&lr=&id=->

- 1rVDwAAQBAJ&oi=fnd&pg=PP1&dq=metodologi+penelitian+&ots=9Y8W-j2SU4&sig=1N80Ap4eTRLvsNyO85ZEpSzU7Qc
- Nasution, F. R., & Ahyaningsih, F. (2024). Implementasi Strategi *Algoritma greedy* Dalam Menyelesaikan Integer Knapsack Problem Pada Perusahaan Jasa Pengiriman Barang PT. Tri Adi Bersama (Anteraja) Kota Medan. *Innovative: Journal Of Social Science Research*, 4(5), 5018–5038.
- Piliang, F. (2017). Penerapan Metode Greedy Knapsack Dalam Menentukan Komposisi Buah Pada Masalah Keranjang. *Jurnal Teknologi Informasi*, 10(2). <https://journal.ubm.ac.id/index.php/teknologi-informasi/article/view/335>
- Rianti, S., Lubis, H., & Pahlevi, R. (2020). Sistem Penunjang Keputusan Optimalisasi Barang Dengan *Algoritma greedy* Pada PT Sentralindo Teguh Gemilang. *JSI (Jurnal Sistem Informasi) Universitas Suryadarma*, 7(2), 123–132. <https://doi.org/10.35968/jsi.v7i2.454>
- Setiawati, S., Fitrawati, E., Awwaliyah, R. R., Alivia, B. N., & Bahri, S. (2024). Application of Greedy Algorithm to Solve Integer Knapsack Problem (Case Study: Indah Logistik Cargo Mataram). *Semeton Mathematics Journal*, 1(1), 33–42. <https://doi.org/10.29303/semeton.v1i1.210>