

HÖRİSTİK YAKLAŞIM

Araş. Gör. H. Murat Ersen^(*)

1. GİRİŞ

Son yıllarda giderek daha yoğun şekilde kullanılan hōristik yaklaşımlar, pek çoklarınınca bilinmemekte veya yanlış anlamlarda kullanılmaktadır. Bu makalede referanslarda belirtilen temel makalelerden yararlanılarak konuya bir açıklık getirilmeye çalışılmış ve yazarın pratikte de uğraştığı ders programının yapılması süreci bir örnek uygulama olarak sunulmuştur.

2. TANIM

Hōristikler, sezgisel bir tercihle, basit ve kısa yoldan sonuca ulaştıran, fakat optimum sonuca yaklaşıldığını garanti etmeyen iteratif algoritmalarıdır.

3. KULLANIM ALANI

Hōristik uygulamaların en çok önem kazandığı alan kombinatoryal problemlerdir. Kombinatoryal problemler bir ya da daha fazla ayrı amaçlar setiyle ilgilenirler. Üç ana gruba ayrılabilirler: atama (assignment), sıralama (sequencing) ve seçme (selecting) problemleri. Gerçek hayatta karşılaşılan kombinatoryal problemlerin çoğu bu üç bileşeni aynı anda içerir.

- i) Atama problemleri
 - Lineer Atama Problemi
 - Transportasyon Problemi
 - Kuadratik Atama Problemi

- ii) Sıralama problemleri
 - En Kısa Yol Problemi
 - Nakit Arbitraji Problemi
 - Gezen Satıcı Problemi

^(*) Hacettepe Üniversitesi İ.İ.B.F. İşletme Bölümü Araştırma Görevlisi

- Çinli Postacı Problemi

iii) Seçme problemleri

- En Az Dallanan Ağaç Problemi
- Köşeleri Kapsama Problemi
- Nodları Kapsama Problemi
- Set Kapsama Problemi
- Knapsack Problemi

Her üç bileşeni de içeren problemlerin standart tiplerine özel isimler verilmiştir:

- Araç Programlama Problemleri
- Üretim ve Akım Planlama Problemleri
- Montaj Hattı Dengeleme Problemleri
- Ders Programı Hazırlama Problemleri
- İş Programı Hazırlama Problemleri

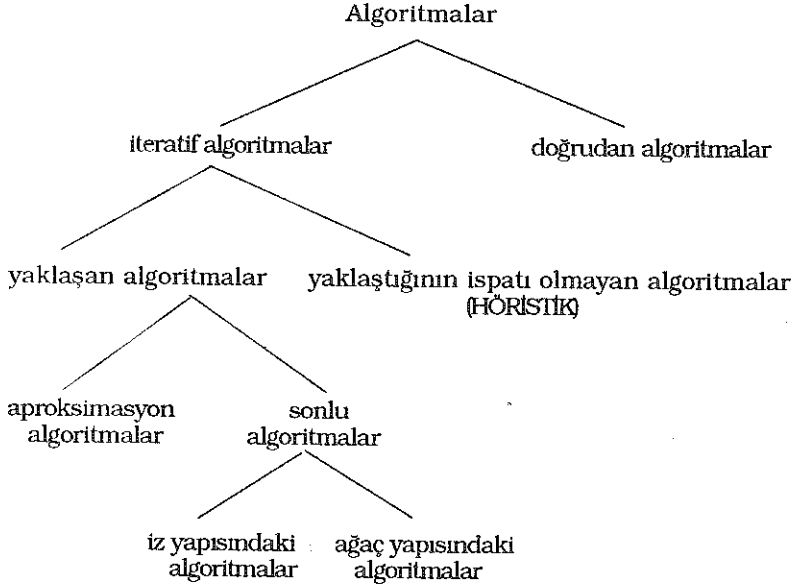
4. KULLANIM NEDENLERİ

Höristik çözüm metodlarının kullanımını gerektirebilecek sebepler şöyle sıralanabilir:

- i. Analitik ya da iteratif bir çözüm prosedürü bilinmemektedir.
- ii. Analitik veya iteratif çözüm prosedürü olsa bile hesaplama güçlüğü veya gerçekçi olmayan veri gereksinimi nedeniyle kullanımı olanaksızdır.
- iii. Höristik metodun anlaşılması tasarımı itibarıyla kullanıcı için kolay olduğundan uygulama şansı yüksek olabilir.
- iv. Optimal olarak çözülebilen bir problemde öğrenme amaçlı kullanılabilir. (Hangi değişkenlerin önemli olduğuna dair sezgi geliştirmek için.)
- v. Optimal çözümü garanti eden iteratif bir prosedürün bir parçası olarak kullanılabilir. (Uygun bir başlangıç çözümü elde etmek veya kesin sonucu verecek bir prosedürün ara adımında karar vermek için.)
- vi. Nümerik bir yaklaşımda hesaplama çabalarını önemli ölçüde azaltacak iyi bir başlangıç çözümü bulmak için kullanılabilir.

5. HÖRİSTİK ALGORİTMALARIN YAPISI

Algoritmalar, matematik terimleriyle ifade edilmiş bir problemin çözümünde kullanılan süreçlerdir. Hüristikler de algoritmaların bir alt kümesidir.



Algoritmalar sadece matematiksel modellere uygulanabilir. Bu nedenle çözülecek problemin çok iyi tanımlanmış olması ve bir matematiksel modelle ifade edilmesi gerekir.

İteratif algoritmaların ortak özelliği belirli bir prosedürün birçok defa tekrarlanmasıdır. Her iterasyon bir çözüm durumundan (solution state) başlar. Burada, bir sonraki iterasyonun çözüm durumuna ulaşabilmek için, aday çözüm altkümeleri belirlenir ve değerlendirilir. Bu adaylar seti, içinde bulunan durumun komşuluğu diye adlandırılır. Bir çözüm durumunun tek bir komşuluğu yoktur. Pek çok iteratif algoritma için bir komşuluklar hiyerarşisi belirlenebilir. Bu hiyerarşi aşağıdaki dört seviyede genellenebilir:

-Komşuluk 1: Potansiyel adaylar seti.

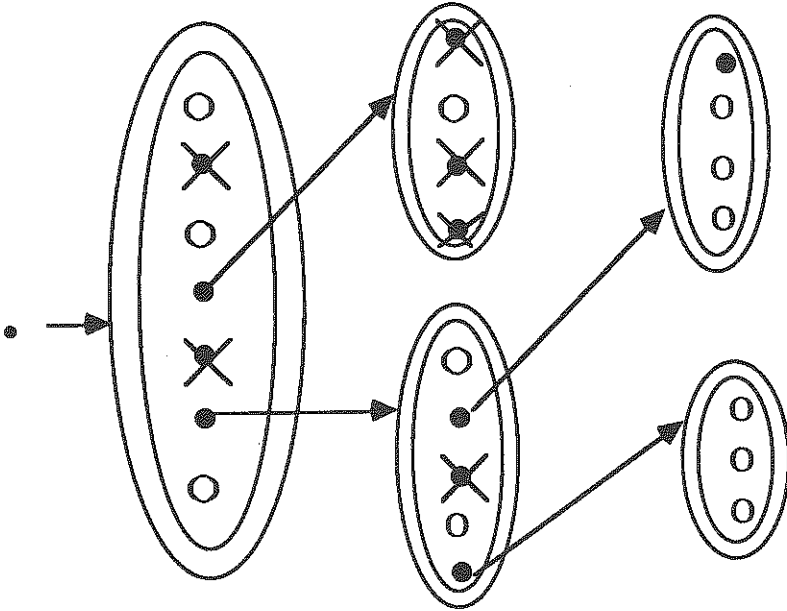
-Komşuluk 2: Gözönüne alınan adaylar (Komşuluk 1'in altkümesi). Komşuluk 2'yi belirlemek için, işe yaramadığı açık olan potansiyel adaylar elenir.

-Komşuluk 3: Kabul edilmiş adaylar seti (Komşuluk 2'nin altkümesi). Komşuluk 3'ü belirlemek için gözönüne alınan adayların hepsi değerlendirilir, çözüm için gerekli olmayanlar atılır.

-Komşuluk 4: Seçilmiş adaylar (Komşuluk 3'ün altkümesi). Komşuluk 4 sadece hüristik için tanımlanmıştır. Kabul edilmiş adayların bazıları özel hüristik kuralları gereğince elenir, kalanı seçilmiş adaylar setini oluşturur. Burada aranan çözümü sağlayacak kümenin elenmesi olasılığı da vardır.

Aşağıdaki figürde dış elips potansiyel adayları (komşuluk 1), iç elips ve içindeki noktalar gözönüne alınan adayları (komşuluk 2), boş noktalar elenen, dolular kabul edilen adayları (komşuluk 3) göstermektedir. Kabul edilen adayların elenenleri çarpı ile işaretlidir. Kalanlar seçilmiş adayları (komşuluk 4) oluşturmaktadır.

Bulunan çözümün kalitesi ve araştırma maliyeti arasında mantıklı bir denge bulunmalıdır. Hüristikler optimum çözüme yaklaştığının ispatı olmayan algoritmalarlardır.



Höristikler iz yapısıyla sona eren indirgenmiş ağaç yapısındadır. Burada, seçilmeyen kümenin daha iyi sonuç verme riski vardır.

6. HÖRİSTİK SÜREÇ İÇİN BAŞARI KOŞULLARI

Höristik çözüm prosedürünün işlemesi için şu üç bileşene gereksinim vardır:

1. Potansiyel çözümlerin alt kümelerini ifade eden sembol yapısı: kod veya veritabanı.

2. Daha rafine potansiyel çözüm setleri elde etmek için veritabanındaki sembolleri modifiye eden işlemler seti: üretim kuralları.

3. Belirli, istenen bir anda veritabanına hangi işlemin uygulanacağına karar veren bir araştırma prosedürü: kontrol stratejisi.

Bir kontrol stratejisi sistematik olmalıdır. Bu da şu şartların sağlanmasıyla mümkün olabilir:

i. Altına bakılmadık hiçbir taş kalmasın (altında birşeyin olmadığına emin oldukların dışında)

ii. Hiçbir taşın altına bir defadan fazla bakılmasın.

7. HÖRİSTİK YAKLAŞIMDA STRATEJİLER

Höristik prosedürlerde temel olarak üç strateji kullanılır:

1. Kısıtları-sağlama.
2. Sıralamayı-sağlama.
3. Strateji-oluşturma.

İlk iki strateji, bir başlangıç pozisyonundan üretim kuralları ile elde edilmiş tüm problem alt setini içeren bir durum grafiğinde yol bulma esasına dayanır.

Üçüncü strateji ise problemin herbiri ayrı ve bağımsız çözümler gerektiren alt problemlere ayrılması esasına dayanır. Araştırma stratejisinin amacı çözüm grafiğini mümkün olan en az dallanma ile oluşturmaktır. Bu tür höristikler daha çok yapay zeka çalışmaları ile ilgili olarak kullanılmaktadır.

8. BİR HÖRİSTİK UYGULAMASI

8.1. PROBLEMİN TANIMI

Bir üniversite ders programının hazırlanması gerçekleştirilmiştir. Program, öğretim görevlilerinin kısıtlarını, bir sınıfın derslerinde ve birbirini izleyen sınıfların derslerinde (herhangi bir dersi tekrar etmek durumunda olan üst sınıf öğrencileri gözönüne alınmıştır) çakışma olmamasını sağlama amacına yöneliktir.

8.2. ÇÖZÜM YAKLAŞIMI

Problemin çözümünde hüristik tekniklerinden yararlanılmıştır. Problemin tanımında belirlenen amaçlar doğrultusunda, öğretim görevlilerine kısıtlarının niteliğine göre öncelikler tanımlanarak, dersler atanmıştır. Bu atama işleminde yerleştirilen her ders kendinden sonra yerleştirilecek dersler için kısıtlamalar getirmektedir. Öncelik sırasına göre yerleştirilirken öğretim görevlisinin tercih ettiği zamanın dolu olması halinde ders, kısıtlamalara uyan ilk boş yere atanmaktadır. Çözüme başlanırken ilk şart olarak derslik sayısının gereksinimi karşılayacak çoklukta olması gerektiği belirlenmiştir. Algoritma, bir sınıfa ait haftalık maksimum yedi ders için geçerlidir. Bu, birbirini izleyen sınıflarda derslerin çakışmaması amacı gözönüne alındığında, haftada maksimum onbeş ders bloğu bulunmasının getirdiği kaçınılmaz bir kısıttır.

8.3. MATEMATİKSEL MODELİN AÇIKLANMASI

Haftalık programda her zaman bloğu, bir numara ile belirlenmiştir. Bir gün üç bloktur. Pazartesi ilk ders 1 olmak üzere cuma günü son ders 15'e kadar sıralanmaktadır.

(i,j) : i SINIF (i : 1,2,3,4)

j DERS (j : 1,2,3,4,5,6,7)

D : Derslik sayısı

olarak belirlendiğinde

$$15 D \geq i * j \quad \text{KAPASİTE KISITI}$$

ilk başta sağlanmalıdır.

8.3.1. Amaçlar

1. Bir öğretim görevlisinin aynı saatte iki dersi olmaması.
2. Bir (i,j) dersinin $(i,j +/- n)$ dersiyile çakışmaması.
3. Bir (i,j) dersinin $(i + 1,j)$ ve $(i - 1,j)$ dersiyile çakışmaması.

8.3.2. Yöntem

Belirlenmesi gereken sabitler: öğretim görevlisi sayısı ($\# H$), öncelik sırası ve hangi dersin hangi öğretim görevlisi tarafından verildiği ve tercih edilen ders zamanlarıdır. Buna göre bağımsız değişkeni zaman (Z), bağımlı değişkeni öncelik sırasına göre yerleştirilmiş öğretim görevlileri (H) olan analitik düzlemde, dersler (i,j) , kısıtlar gözönüne alınarak yerleştirilir. Atama öncelik sırasına göre yapılır; atama yapılacak yerin dolu olması halinde zaman sırasında ilk uygun boşluğa atama yapılır.

8.3.3. Kısıtlar

Kısıtlar, amaçlar doğrultusunda ilk atamanın yapılmasıyla ortaya çıkmaktadır. Bir (i,j) dersinin atanması, haftalık program (i,j) 'lerden oluşan bir matris olarak düşünülürse, o dersin bulunduğu zaman sütununu diğer (i,j) ve $(i - 1,j)$, $(i + 1,j)$ dersleri için kapatmaktadır.

8.3.4. Girdiler

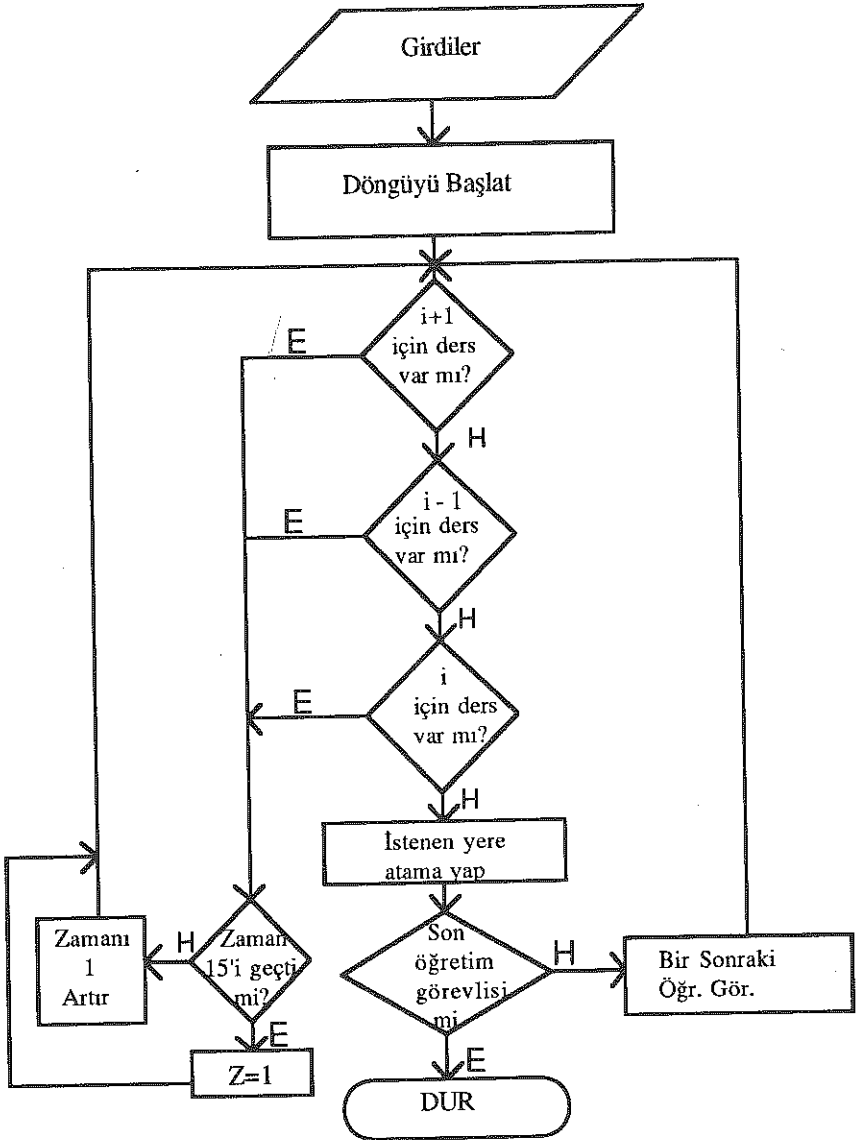
Öğretim görevlisi sayısı.

Hangi dersin hangi öğretim görevlisi tarafından verildiği.

Öğretim görevlilerinin öncelik sırası.

Öğretim görevlilerinin zaman tercihleri.

8.3.5. Akım Şeması



9. SONUÇ

Bu çalışmada hüristik yaklaşımların tanımı, kullanım alan ve sebepleri, algoritmik yapıları, hüristik süreç içinde başarı koşulları ve hüristik yaklaşım stratejileri tartışılmıştır. Bir örnek uygulama çerçevesinde haftalık ders programının yapılması gerçekleştirilmiştir.

Bu uygulamada kısıtlamaları-sağlayan hüristik tekniğinden faydalanılmış, kişisel deneyimlerle kazanılan sezgi yol gösterici olmuştur. Ulaşılmak istenen amaçlar, kısıtlar ve yöntem tartışılabilir. Fakat bu uygulama ders programı yapılması sorununa bir çözüm getirmekten ziyade hüristik tekniğinin uygulamasını göstermeye yöneliktir. Bu nedenle başka kriterler gözününe alınarak daha gelişmiş, daha iyi çözümler bulunabilir. Vurgulanması gereken; sezgisel bir tercihle, çok kısa sürede sonuca ulaşılmasıdır.

KAYNAKÇA

1. Müller-Merbach, H., "Heuristics and Their Design: a Survey", European Journal Of Operational Research, 8 (1981) 1-23.
2. Silver, E.A., Vidal, R.V.V., ve Werra de, D., "A Tutorial On Heuristic Methods", European Journal Of Operational Research, 5 (1980) 153-162.
3. Simon, H.A., ve Newell A., "Heuristic Problem Solving: The Next Advance in Operations Research" Operations Research Jan-Feb 1958.
4. Tonge, F.M., "The Use of Heuristic Programming in Management Science", 7th Annual International Meeting of The Institute of Management Sciences, New York, October 21, 1960.
5. Zanakis, S.H., ve Evans, J.R., "Heuristic Optimization: Why, When, and How To Use It" Interfaces Vol.11, No.5, October 1981.