الگوریتم جستجوی هارمونیک (Harmony Search):

این الگوریتم الهام گرفته از فرآیند بداهه نوازی موسیقیدانان است. در ادامه، مراحل اجرای این الگوریتم به طور مختصر توضیح داده می‌شود. این توضیحات برای کسانی که با الگوریتم‌های بهینه‌سازی آشنایی دارند بسیار مفید خواهد بود.

مراحل HS:

۱- پارامترهای قابل تنظیم الگوریتم مقدار دهی اولیه می‌شوند.

Harmony Memory Size (HM) اندازه‌ی حافظه‌ی هارمونی

Harmony Memory Considery Rate (HMCR) آهنگ لحاظ کردن حافظه‌ی هارمونی

Pitch Adjusting Rate (PAR) آهنگ تنظیم پیچ

Band Width of Generation (bw) پهنای باند تولید

T_{max} بیشترین تعداد تکرار

۲- مقدار دهی اولیه به HM (HMSتا هارمونی برای HM تولید می‌شود)

$x_i(j)=l(j)+rand*(u(j)-l(j))$

$i=1,2,\ldots,HMS$

$j=1,2,\ldots,d$ (بعد مسئله)

(l,u نشان دهنده‌ی مینیمم و ماکزیمم مربوط به هر متغییر تصمیم مسئله می‌باشد)

۳- مقدار تابع هدف برای هر هارمونی محاسبه می‌شود.

۴- هارمونی جدید طبق الگوی زیر بداهه نوازی یا تولید می‌شود.

```
for j=1:d
 if r1>=HMCR
```

$x_{new}(j)=l(j)+r2*(u(j)-l(j))$

```
else
```

n_{HM}=(یک عدد صحیح تصادفی بین ۱و اندازه‌ی حافظه‌ی هارمونی)

$x_{new}(j)=HM(n_{HM},j)$

```
if r3<PAR
```

$x_{new}(j)=x_{new}(j)+(r4-r5)*bw*(u(j)-l(j))$

```
end
```

matlabproject.ir

```
end
end
```

(r1، r2، r3، r4 و r5 اعدادی تصادفی بین صفر و یک هستند)

۵- اگر هارمونی جدید در فضای جستجو باشد و مقدار تابع هدف آن بهتر از مقدار تابع هدف بدترین هارمونی HM باشد، جایگزین بدترین هارمونی HM می‌شود، در غیر اینصورت رها می‌شود.

۶- مراحل ۳ تا ۵ تا زمان برآورده شدن معیار توقف تکرار می‌شوند.

فرمول‌های زیر در اجرای مراحل الگوریتم قابل استفاده هستند

$$PAR(t)=PAR_{min}+(PAR_{max}-PAR_{min})*t/t_{max}$$

$$bw(t)=bw_{max}*exp(c*t)$$

$$C=(ln(bw_{min}/bw_{max}))/(t_{max})$$

matlabproject.ir