Ecological Optimization Using Harmony Search

ZONG WOO GEEM, JUSTIN C. WILLIAMS
Environmental Planning & Management Program
Johns Hopkins University
Baltimore, Maryland, USA

Abstract: The music-inspired meta-heuristic algorithm, harmony search, was applied to a natural reserve selection problem for preserving species and their habitats. The problem was formulated as an optimization problem (maximal covering species problem; MCSP) to maximize covered species with minimal efforts. Then, it was solved by an improved harmony search (HS) algorithm which includes problem-specific operations. When applied to real-world problem in the state of Oregon, USA, the harmony search algorithm found better solutions than those of another meta-heuristic algorithm, simulated annealing.

Key-Words: Harmony search, Optimization, Maximal covering species problem, Evolutionary algorithm, Meta-heuristics, Soft computing

1 Introduction

In modern industrial and urbanized life, conserving ecosystem and its species is very important. In order to do so, quantitative optimization techniques have been developed and utilized for the nature reserve site selection problem.

In the light of the optimization, ReVelle et al. [1] reviewed five classes of the reserve selection problem: (1) species set covering problem (SSCP); (2) maximal covering species problem (MCSP); (3) backup and redundant covering problem (M Multiple-Representation Species Problem; MMRSP); (4) chance constrained covering problem; and (5) expected covering problem. Out of the above-mentioned five classes, the MCSP was especially tackled by various algorithms [2-4].

In this study, we apply a recently-developed harmony search (HS) algorithm to the MCSP and to compare our results with those of another meta-heuristic algorithm, simulated annealing (SA), from the literature [2].

2 Problem Formulation

Of the five classes of reserve selection problems discussed in [1], SSCP and MCSP are especially popular forms. The SSCP is to find the least number of parcels while covering every species.

The mathematical formulation of the SSCP model is as follows:

$$
\text{Min } \sum_{j \in J} x_j
$$  (1)

$$
\text{s.t. } \sum_{j \in M_i} x_j \geq 1, \text{ all } i \in I. 
$$  (2)

where $j$ and $J$ are the index and set of land parcels, respectively; $i$ and $I$ are the index and set of species, respectively; $M_i$ is the set of parcels $j$ that include species $i$; and $x_j$ is a binary variable for parcel selection (it has 1 if parcel $j$ is selected, and has 0 otherwise).

The MCSP is to find the maximal number of species while limiting the number of selected parcels to $P$. The mathematical formulation of the MCSP model is as follows:

$$
\text{Max } \sum_{i \in I} y_i
$$  (3)

$$
\text{s.t. } \sum_{j \in M_i} x_j \geq y_i, \text{ all } i \in I, 
$$  (4)

$$
\sum_{j \in J} x_j = P.
$$  (5)

where $y_i$ is a binary variable for species covering (it has 1 if species $i$ is covered, and has 0 otherwise).
3 Harmony Search Algorithm
The HS algorithm originally came from the analogy between music improvisation and optimization process [5]. This algorithm has been successfully applied to various discrete optimization problems such as traveling salesperson problem [5], tour routing [6], music composition [7], Sudoku puzzle solving [8], water network design [9], dam operation [10], vehicle routing [11], and structural design [12].

The HS algorithm searches for optimal solution vectors using a novel stochastic derivative as in Equation 6 [13], which is based on the density information of multiple solution vectors, instead of gradient information of single solution vector. In other words, if a certain value frequently appears in multiple vectors, the value has higher chance to be selected.

\[
\frac{\partial f}{\partial x_i} = \frac{1}{K_i} \cdot P_{Random} + \frac{n(x_i(k))}{HMS} \cdot P_{Memory} + \frac{n(x_i(k + m))}{HMS} \cdot P_{Pitch} \tag{6}
\]

The stochastic derivative in Equation 6 stands for the total probability (random selection probability + memory consideration probability + pitch adjustment probability) to select a certain value \(x_i(k)\) from candidate discrete value set \(\{x_i(1), x_i(2), ..., x_i(K_i)\}\) or from a harmony memory (HM) \(\{x_i^1, x_i^2, ..., x_i^{HMS}\}\) which stores multiple solution vectors. Here, \(f(\cdot)\) is an objective function; \(n(\cdot)\) is a frequency function which counts the number of a certain value \(x_i(k)\) in the HM; \(k\) is an index; \(K_i\) is the number of candidate values for the variable \(x_i\); \(m\) is a neighboring index and has normally 1; and HMS is the number of multiple vectors stored in the HM.

The first term in the right hand side in Equation 6 indicates the probability to randomly select \(x_i(k)\) from candidate value set with the probability of \(P_{Random}\); the second term the probability to select \(x_i(k)\) from the HM with the probability of \(P_{Memory}\); the third term the probability to select \(x_i(k)\) after its neighboring values are selected from the HM, with the probability of \(P_{Pitch}\).

The basic procedure of HS is as follows:

- Step 0. Randomly generate multiple solution vectors as many as HMS (harmony memory size).
- Step 1. Generate a new vector based on the stochastic derivate in Equation 6.
- Step 2. If the new vector is better than the worst one in the HM in terms of an objective function, the new one is included in the HM and the worst one is excluded from the HM.
- Step 3. Repeat Steps 1 and 2 until stopping criteria is satisfied.

For example, consider a minimization problem as follows:

\[
\text{Min } f(x_1, x_2) = (x_1 - 2)^2 + (x_2 - 3)^2 \tag{7}
\]

In Step 0, two vectors (3, 2) and (4, 5) are randomly generated with the candidate value set of \(\{1, 2, 3, 4, 5\}\). The HM can be as follows:

\[
\text{HM} = \begin{bmatrix}
3 & 2 & 1 \\
4 & 5 & 8
\end{bmatrix} \tag{8}
\]

In Step 1, the value of \(x_1^{\text{New}}\) can be 4 if it is chosen from the HM \(\{3, 4\}\); and the value of \(x_2^{\text{New}}\) can be 3 if 2 is first chosen from the HM \(\{2, 5\}\), then 2 is pitch-adjusted into a neighboring value 3. The new vector (4, 3) has the objective function value of 4.

In Step 2, the new vector (4, 3) is included in the HM and the worst vector (4, 5) is excluded from the HM because the former has better value than the worst one.

4 Application
The HS model developed for the MCSP is tested with Oregon data, which consists of 426 species and 441 parcels. Figure 1 shows the hex map of the state of Oregon.

For being applied to the MCSP, the structure of the original HS algorithm is modified. Normally pitch adjustment operation is important because it helps the algorithm to locally search. However, because the
decision variable \( x_j \) has only two candidate values \( \{0, 1\} \), it (third term in Equation 6) is omitted.

Also, because the value of \( P \) (which ranged from 1 to 24) in Equation 5 is relatively small when compared with the number (441) of entire variables, the decision variable \( x_j \) should have 1 with the following probability, instead of 50 \%, in initial multiple vector generation (Step 0) and random selection operation (first term in Equation 6):

\[
x_j \leftarrow \begin{cases} 
1 \quad \text{w.p. } \frac{N_j}{\sum_j N_j} , \quad j = 1, \ldots, J \\
0 \quad \text{otherwise} 
\end{cases} 
\]  (9)

where \( N_j \) is the number of species that are included in parcel \( j \); and \( N \) is the number of total species (= 426 in this example) eligible to be covered.

Equation 9 allows us to efficiently generate a proper solution vector in sparse selection. In other words, it helps the algorithm to choose only 1 to 24 parcels for a solution vector which has 441 candidate parcels.

However, some vectors have selections (1’s) less than \( P \) times. For this reason, a coefficient \( \alpha \) (it has 1.5 in this study). The numerical results with 1.5 were better than those with 1.0) is added to Equation 9 as follows:

\[
x_j \leftarrow \begin{cases} 
1 \quad \text{w.p. } \alpha \left( \frac{N_j}{\sum_j N_j} \right) , \quad j = 1, \ldots, J \\
0 \quad \text{otherwise} 
\end{cases} 
\]  (10)

If the number of selected parcels reaches \( P \), HS stops considering additional selection using Equation 10 and goes to Step 2 in the HS procedure. In order not to lose parcels which have more species than others, each decision variable is considered in descending order of selection rate in Equation 10, instead of sequential order of \( j \).

In order to start with better initial vectors in Step 0, the HM is filled with chosen multiple vectors, as many as HMS, after \( n \times \text{HMS} \) vectors are generated. This study uses the value of 5 as \( n \) after several trials. In addition, the number of identical vectors in the HM is limited to prevent the premature HM which has many local optima. This study uses the value of 2 as the number of maximum allowed identical vectors because it was frequently used in the literature.

For pursuing the diversity of vectors in the HM, if a parcel contains a species which is not yet included in the HM, it is deterministically selected regardless of Equation 10 every tenth iteration.

For the MCSP, the HS algorithm approached 24 different \( P \) cases \( (P = 1, 2, \ldots, 24) \) as shown in Table 1. The HS results were also compared with those of another meta-heuristic algorithm, SA [2] as well as those of exact method [3]. When compared with SA, HS found better solutions in 14 cases \( (P = 8, 11, 12, 13, 14, 15, 17, 18, 19, 20, 21, 22, 23, 24) \) while it found worse solution only once \( (P = 9) \). When compared with exact method, HS found optimal solutions or near-optimal solutions with maximum one species gap.

In each \( P \) case, the HS used various HMS values \( \{10, 30, 50, 100\} \) and \( P_{\text{Memory}} \) values \( \{0.9, 0.95, 0.97\} \). Here, \( P_{\text{Random}} \) is \( (1 - P_{\text{Memory}}) \).

Table 2 shows the results with various algorithm parameter values when \( P = 24 \). HS found the best solution (426) four times out of 12 cases, ranging 421 to 426. The last column shows the iteration at which the HS found the solution. This study tried up to 30,000 iterations, taking up to 34 minutes per each run with MS Excel.
Table 1. Comparison of MCSP Results

<table>
<thead>
<tr>
<th>P</th>
<th>HS</th>
<th>SA</th>
<th>Opt</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>254</td>
<td>254</td>
<td>254</td>
</tr>
<tr>
<td>2</td>
<td>318</td>
<td>318</td>
<td>318</td>
</tr>
<tr>
<td>3</td>
<td>356</td>
<td>356</td>
<td>356</td>
</tr>
<tr>
<td>4</td>
<td>374</td>
<td>374</td>
<td>374</td>
</tr>
<tr>
<td>5</td>
<td>384</td>
<td>384</td>
<td>384</td>
</tr>
<tr>
<td>6</td>
<td>390</td>
<td>390</td>
<td>390</td>
</tr>
<tr>
<td>7</td>
<td>395</td>
<td>395</td>
<td>395</td>
</tr>
<tr>
<td>8</td>
<td>400</td>
<td>398</td>
<td>400</td>
</tr>
<tr>
<td>9</td>
<td>402</td>
<td>403</td>
<td>403</td>
</tr>
<tr>
<td>10</td>
<td>405</td>
<td>405</td>
<td>406</td>
</tr>
<tr>
<td>11</td>
<td>408</td>
<td>407</td>
<td>408</td>
</tr>
<tr>
<td>12</td>
<td>410</td>
<td>409</td>
<td>410</td>
</tr>
<tr>
<td>13</td>
<td>412</td>
<td>411</td>
<td>412</td>
</tr>
<tr>
<td>14</td>
<td>414</td>
<td>413</td>
<td>414</td>
</tr>
<tr>
<td>15</td>
<td>416</td>
<td>415</td>
<td>416</td>
</tr>
<tr>
<td>16</td>
<td>417</td>
<td>417</td>
<td>418</td>
</tr>
<tr>
<td>17</td>
<td>419</td>
<td>418</td>
<td>419</td>
</tr>
<tr>
<td>18</td>
<td>420</td>
<td>419</td>
<td>420</td>
</tr>
<tr>
<td>19</td>
<td>421</td>
<td>420</td>
<td>422</td>
</tr>
<tr>
<td>20</td>
<td>422</td>
<td>421</td>
<td>423</td>
</tr>
<tr>
<td>21</td>
<td>423</td>
<td>422</td>
<td>424</td>
</tr>
<tr>
<td>22</td>
<td>424</td>
<td>423</td>
<td>425</td>
</tr>
<tr>
<td>23</td>
<td>425</td>
<td>424</td>
<td>426</td>
</tr>
<tr>
<td>24</td>
<td>426</td>
<td>425</td>
<td>NA</td>
</tr>
<tr>
<td>25</td>
<td>NA</td>
<td>426</td>
<td>NA</td>
</tr>
</tbody>
</table>

Table 2. Results with Various Parameter Values

<table>
<thead>
<tr>
<th>HMS</th>
<th>P&lt;sub&gt;Memory&lt;/sub&gt;</th>
<th>Solution</th>
<th>Iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.90</td>
<td>425</td>
<td>6,783</td>
</tr>
<tr>
<td></td>
<td>0.95</td>
<td>424</td>
<td>17,298</td>
</tr>
<tr>
<td></td>
<td>0.97</td>
<td>421</td>
<td>1,557</td>
</tr>
<tr>
<td>30</td>
<td>0.90</td>
<td>425</td>
<td>24,220</td>
</tr>
<tr>
<td></td>
<td>0.95</td>
<td>426</td>
<td>5,675</td>
</tr>
<tr>
<td></td>
<td>0.97</td>
<td>424</td>
<td>9,049</td>
</tr>
<tr>
<td>50</td>
<td>0.90</td>
<td>425</td>
<td>15,336</td>
</tr>
<tr>
<td></td>
<td>0.95</td>
<td>425</td>
<td>4,356</td>
</tr>
<tr>
<td></td>
<td>0.97</td>
<td>426</td>
<td>10,266</td>
</tr>
<tr>
<td>100</td>
<td>0.90</td>
<td>425</td>
<td>28,943</td>
</tr>
<tr>
<td></td>
<td>0.95</td>
<td>426</td>
<td>26,389</td>
</tr>
<tr>
<td></td>
<td>0.97</td>
<td>426</td>
<td>10,778</td>
</tr>
</tbody>
</table>

Table 3 shows alternative solutions the HS found (P = 24 and # of covered species = 426). If certain reserve is not available in one solution, other alternative solutions that do not contain the specific reserve can be considered. While interchange heuristic [3] found three alternative solutions, the HS found 25 alternative solutions.

Table 3. Alternative Solutions of HS

<table>
<thead>
<tr>
<th>Solutions (P = 24, # of covered species = 426)</th>
</tr>
</thead>
<tbody>
<tr>
<td>9 11 24 26 55 75 102 114 120 136 147 169 222 224 289</td>
</tr>
<tr>
<td>314 319 324 345 357 367 375 428 440</td>
</tr>
<tr>
<td>9 11 24 26 55 75 102 120 128 136 147 169 222 224 289</td>
</tr>
<tr>
<td>314 319 324 345 357 367 375 428 440</td>
</tr>
<tr>
<td>9 11 24 26 55 75 102 120 136 147 169 222 224 289</td>
</tr>
<tr>
<td>314 319 324 345 357 367 375 428 440</td>
</tr>
<tr>
<td>9 11 24 27 55 75 120 121 134 147 169 222 225 289</td>
</tr>
<tr>
<td>313 314 319 324 364 375 386 428 440</td>
</tr>
<tr>
<td>9 11 24 27 55 75 120 121 134 147 169 222 225 289</td>
</tr>
<tr>
<td>313 314 319 324 364 375 386 428 440</td>
</tr>
<tr>
<td>9 11 24 27 55 75 120 121 134 147 169 222 225 289</td>
</tr>
<tr>
<td>313 314 319 324 364 375 386 428 440</td>
</tr>
<tr>
<td>9 11 24 27 55 75 120 121 134 147 169 222 225 289</td>
</tr>
<tr>
<td>313 314 319 324 364 375 386 428 440</td>
</tr>
<tr>
<td>9 11 24 27 55 75 120 121 134 147 169 222 225 289</td>
</tr>
<tr>
<td>313 314 319 324 364 375 386 428 440</td>
</tr>
<tr>
<td>9 11 24 27 55 75 120 121 134 147 169 222 225 289</td>
</tr>
<tr>
<td>313 314 319 324 364 375 386 428 440</td>
</tr>
</tbody>
</table>
5 Conclusions

The HS algorithm was applied to an ecological optimization problem (MCSP) where the number of preserved species in an area is to be maximized while limiting the number of considered area parcels.

The HS model in this study was modified from its original structure in order to be applied to MCSP:

- Because the candidate value of each variable is 0 or 1, the HS model does not have pitch adjustment operation.
- Because each solution vector is sparse (many 0’s and few 1’s), the chance to be selected is limited (much lower than 50%) based on the number of species in each area parcel.
- In order to start with good environment, solution vectors in initial HM are chosen from \( n \)-times generations.
- For the diversity of solution vectors in HM, a vector which has new species is deterministically included in the HM.

The HS model was applied to real-world problem (the state of Oregon) which has 426 species and 441 parcels. In 24 cases with different parcel number, the HS found global optimum solutions in 15 cases and near-optimal solutions (only one species gap) 9 cases.

When compared with other meta-heuristic algorithm, the HS algorithm found better solutions than those of the SA algorithm in 14 cases while the former found worse solution only once.

Another advantage of the HS algorithm is the fact that it suggests many alternative solutions because it simultaneously handles multiple solution vectors. For example, the HS found 25 different alternative solutions for the case of \( P = 24 \).

If HS considers additional problem-specific heuristics as well as basic operators, the authors expect that it will perform better than the proposed HS model in this study.

References: