COMBINED ECONOMIC AND EMISSION DISPATCH USING EVOLUTIONARY ALGORITHMS-A CASE STUDY

A. Lakshmi Devi and O. Vamsi Krishna
Department of Electrical and Electronic Engineering, SVUCE, Tirupati, A. P., India
E-mail: energylak@yahoo.co.in

ABSTRACT
An efficient and optimum economic operation of electric power generation systems has always occupied an important position in the electric power industry. This involves allocation of the total load between the available generating units in such a way that the total cost of operation is kept at a minimum. In recent years this problem has taken a suitable twist as the public has become increasingly concerned with environmental matters, so that economic dispatch now includes the dispatch of systems to minimize pollutants, as well as to achieve minimum cost. This paper proposes a lambda based approach for solving the Combined Economic and Emission Dispatch (CEED) problem using Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) methodologies considering the power limits of the generator. The purpose of Combined Economic and Emission Dispatch (CEED) is to minimize both the operating fuel cost and emission level simultaneously while satisfying load demand and operational constraints. This multi-objective CEED problem is converted into a single objective function using a modified price penalty factor approach.

Keywords: combined economic and emission dispatch, CEED, genetic algorithm, population, particle swarm optimization, particles.

INTRODUCTION
In recent years the economic dispatch problem has taken a suitable twist as the public has become increasingly concerned with environmental matters. The absolute minimum cost is not any more the only criterion to be met in the electric power generation and dispatching problems. The generation of electricity from the fossil fuel releases several contaminants, such as sulfur oxides (SO2), and oxides of nitrogen (NOx) into the atmosphere. These gaseous pollutants cause harmful effects on human beings as well as on plants and animals. The Clean Air Act Amendments of 1990 (CAAA) mandates that the electric utility industry should reduce its SO2 emission by 10 million ton/year and the NOx by 2 million ton/year from the 1980 level [4].

The limiting levels of emissions over a schedule horizon represent additional operational constraints that are to be satisfied when finding the optimal solution for the economic dispatch problem. The characteristics of emissions of different pollutants are different and are usually highly non-linear. This increases the complexity and non-monotonicity of the emission constrained economic dispatch problem. This work focuses on emission of nitrogen oxides (NOx) only, because its control is a significant issue at the global level.

PROBLEM FORMULATION
The economic dispatch and emission dispatch are considerably different. The economic dispatch deals with only minimizing the total fuel cost (operating cost) of the system violating the emission constraint. On the other hand emission dispatch deals with only minimizing the total emission of NOx from the system violating the economic constraints. Therefore it is necessary to find out an operating point, that strikes a balance between cost and emission. This is achieved by combined economic and emission dispatch (CEED). The multi-objective combined economic and emission dispatch problem is converted into single optimization problem by introducing price penalty factor h [7]:

Minimize \( \Phi = F + h*E \) (Rs./hr) \hspace{1cm} (1a)

Subject to the constraints:

\[ \sum_{i=1}^{n} P_i = P_D + P_{loss} \] \hspace{1cm} (1b)

and

Max \( (P_{i_{min}}) \leq P_i \leq \text{Min} (P_{i_{max}}) \) \hspace{1cm} (1c)

The price penalty factor \( h \) blends the emission with the normal fuel costs and \( \Phi \) is the total operating cost of the system (i.e., the cost of fuel + the implied cost of emission).

Once the value of price penalty factor is determined, the problem reduces to a simple economic dispatch problem. By proper scheduling of generating units, comparative reduction is achieved in both total fuel cost and NOx emission.

COMBINED ECONOMIC AND EMISSION DISPATCH USING GENETIC ALGORITHM

Genetic string representation
Before applying a GA to any task, a computer compatible representation or encoding must be developed. These representations are referred to as chromosome. The most common representation is a binary string, where sections of the string represent encoding parameters of the solution. The number of bits assigned to a given parameter will determine the numerical accuracy.
Encoding and decoding

Implementation of a problem in a GA starts from the parameter encoding (i.e., the representation of the problem). The encoding must be carefully designed to utilize the GA's ability to efficiently transfer information between chromosome strings and objective function of problem. The proposed approach uses the equal system $\lambda$ (equal system incremental cost) criterion as its basis. The only encoded parameter is the normalized system incremental cost, $\lambda_{nm}$. The advantage of using system $\lambda$ instead of units' output as the encoded parameter is that the number of bits of chromosome will be entirely independent of the number of units. This is particularly attractive in large-scale systems.

The resolution of the solution depends upon the number of bits used to represent $\lambda_{nm}$. In other words, the more encoding bits there are, the higher the resolution. However, on the other hand, the more encoding bits there are, the slower the convergence. In this paper, 12 bits are used to represent $\lambda_{nm}$. Figure-1 shows the encoding scheme of $\lambda_{nm}$.

![Image](image_url)

Figure-1. The encoding scheme of $\lambda_{nm}$

Evaluation of a chromosome is accomplished by decoding the encoded chromosome string and computing the chromosome's fitness value using the decoded parameter. The decoding of $\lambda_{nm}$ can be expressed as:

$$\lambda_{nm} = \sum_{i=1}^{12} (d_i \times 2^{-i}) \quad d_i \in \{0,1\}$$  \hspace{1cm} (2)

The relationship between the actual system incremental cost, $\lambda_{eq}$, and the normalized system incremental cost, $\lambda_{nm}$ is:

$$\lambda_{eq} = \lambda_{nm} \cdot \left( \frac{\lambda_{max} - \lambda_{min}}{\lambda_{max} - \lambda_{eq}} \right)$$  \hspace{1cm} (3)

Where, $\lambda_{min}$ and $\lambda_{max}$ are the minimum and maximum values of system incremental cost.

The fitness function

Implementation of a problem in a genetic algorithm is realized within the fitness function. Since the proposed approach uses the equal incremental cost criterion as its basis, the constraint equation can be rewritten as:

$$\varepsilon = \sum_{i=1}^{n} (P_i - P_D - P_{loss})$$  \hspace{1cm} (5)

Then, the converging rule is when $\varepsilon$ (error) decreases to within a specified tolerance. In order to emphasize the "best" chromosomes and speed up convergence of the iteration procedure, fitness is normalized into the range between 0 and 1. The fitness function adopted is:

$$FIT = \frac{1}{1 + k \left( \frac{\varepsilon}{P_D} \right)}$$  \hspace{1cm} (6)

Where, $k$ is a scaling constant ($k = 50$ in this study).

Genetic operation

Genetic operators are the stochastic transition rules employed by GA. These operators are applied on each string during each generation to generate a new and improved population from the old one. A simple GA consists of three basic operators: Elitism, Crossover and Mutation.

Elitism

The copying of best population to next population is called “Elitism”. If the probability is high, then the convergence rate increases. But it will not be too high to get the good result. The implementation of elitism is done by choosing the best population from the previous generation. The population is chosen as 60 so initially the performance index for all the population is calculated and then the chromosomes are arranged in the descending order according to their fitness value. Then the first 15% of the population is copied to the next generation.

Crossover

Uniform crossover technique is adapted in this problem. For carrying out the crossover, there is a need to identify the parents. The parent selection is done by using the Roulette wheel technique.

This parent selection is to be repeated two times to get the two parents for crossover. After selecting the parents, a random number is generated between 0 and 1, and then this random number is compared with the crossover probability ($P_c$). If it is less than $P_c$, crossover is performed. If it is greater than $P_c$, Par1 and Par2 are directly selected as Child1 and Child2. The crossover probability is taken as 0.70.
Mutation
Mutation is the process of random modification of the value of a string position with a small probability. It is not a primary operator but it ensures that the probability of searching any region in the problem space is never zero and prevents complete loss of genetic material through reproduction and crossover. The mutation probability is taken as 0.01.

Algorithm
The algorithm for solving the combined economic and emission dispatch problem using Genetic Algorithm method is given below:

1. Read generator data, emission data, P limits, B-coefficients, power demand and GA parameters.
2. Compute the modified price penalty factor $h_m$ as per steps discussed above.
3. Generate initial population of chromosome of binary bits using random generation technique.
4. Set the iteration count = 1.
5. Set chromosome count = 1.
6. Decode the chromosomes of the population and determine normalized system incremental cost, $\lambda_{f,e}$.
7. Calculate the actual system incremental cost, $\lambda_{F,e}$ using eqn. (3).
8. Calculate the generation output of all the units for each chromosome from its $\lambda_{F,e}$ value using eqn. (4) and enforce $P_i$ limits.
9. Calculate transmission losses using B-coefficient equation and compute the error using eqn. (5).
10. Calculate the fitness value of the chromosome, using the eqn. (6).
11. Repeat the procedure from step no. 6 until chromosome count > population size.
12. Sort the chromosomes and all their related data in the descending order of fitness.
13. Check if the error (1) is less than $\varepsilon$. If yes, go to 20.
14. Copy the $P_i, %$ chromosomes of old population to new population starting from the best ones from the top.
15. Perform crossover on selected parents and generate new child chromosomes, repeat it to get required number of chromosomes.
16. Add all the generated child chromosomes to new population.
17. Perform mutation on all chromosomes.
18. Replace old population with new population.
19. Increment iteration count. If iteration count < max. iteration, go to 5; else print the message “problem not converged in maximum number of iterations”.
20. Calculate the total fuel cost, total emission release, emission cost etc. Print the result.

COMBINED ECONOMIC AND EMISSION DISPATCH USING PARTICLE SWARM OPTIMIZATION

Overview of Particle Swarm Optimization (PSO)
Like evolutionary algorithms, PSO technique conducts search using a population of particles, corresponding to individuals. Each particle represents a candidate solution to the problem at hand. In a PSO system, particle changes their positions by flying around in a multi-dimensional search space until computational limitations are exceeded. In Particle Swarm Optimization, a particle is defined as a moving point in hyperspace. For each particle, at the current time step, a record is kept of the position, velocity, and the best position found in the search space so far.

Let $x$ and $v$ denote a particle coordinates (position) and its corresponding flight speed (velocity) in a search space, respectively. The best previous position of a particle is recorded and represented as $pbest$. The index of the best particle among all the particles in the group is represented as $gbest$. Each particle knows the best value so far ($pbest$) and best value in the group ($gbest$). The particle tries to modify its position using the current velocity and the distance from $pbest$ and $gbest$. At last, the modified velocity and position of each particle can be calculated as using the following formulas:

$$v_i^{k+1} = w \cdot v_i^k + c_1 \cdot \text{rand}_i \cdot (pbest_i - x_i) + c_2 \cdot \text{rand}_2 \cdot (gbest_i - x_i)$$ (7a)

$$x_i^{k+1} = x_i^k + v_i^{k+1}$$ (7b)

where

- $v_i^k$: velocity of particle $i$ at iteration $k$
- $W$: inertia weight factor
- $c_1, c_2$: learning factor
- $\text{rand}_1, \text{rand}_2$: random number between 0 and 1
- $x_i^k$: position of particle $i$ at iteration $k$

It is worth mentioning that the second term in eqn. (7a) represents the cognitive part of PSO where the particle changes its velocity based on its own thinking and memory. The third term represents the social part of PSO where the particle changes its velocity based on the social-psychological adaptive of knowledge.

The constants $c_1$ and $c_2$ pull each particle towards $pbest$ and $gbest$ positions. Low values allow particles to roam far from the target regions before being tugged back. On the other hand, high values result in abrupt movement towards, or past, target regions. Hence, the acceleration constants $c_1$ and $c_2$ are often set to be 2.0 according to past experiences. Suitable selection of inertial weight ‘$w$’ provides a balance between global and local explorations, thus requiring less iteration on average to find a sufficiently optimal solution. As originally developed, $w$ often decreases linearly from about 0.9 to 0.4 during a run. In general, the inertia weight $w$ is set according to the following equation,
\[ W = W_{\text{max}} - \left( \frac{W_{\text{max}} - W_{\text{min}}}{\text{iter}_{\text{max}}} \right) \times \text{iter} \]  

where

- \text{iter} : current iteration number
- \text{iter}_{\text{max}} : maximum number of iterations

In the iteration process the particle velocity is limited by some maximum value \( v_i^{\text{max}} \). The parameter \( v_i^{\text{max}} \) determines the resolution, or fitness, with which regions are to be searched between the present position and the target position. This limit enhances the local exploration of the problem space and it realistically simulates the incremental changes of human learning. If \( v_i^{\text{max}} \) is too high, particles might fly past good solutions. If \( v_i^{\text{max}} \) is too small, particles may not explore sufficiently beyond local solutions. In many experiences with PSO, \( v_i^{\text{max}} \) was often set at 10-20\% of the dynamic range of the variable on each dimension.

**Constriction factor approach**

This factor may help in sure convergence. A low value of \( k \) facilitates rapid convergence and little exploration - high values gives slow convergence and much exploration. The constriction factor was proposed by the mathematician Maurice Clerc. He has studied about the convergence condition of particle swarm system by means of second order differential equations. In the constriction model \( k \) is set as a function of \( c_1 \) and \( c_2 \), so that convergence is ensured- even without \( v_i^{\text{max}} \). The constriction factor in the velocity update equation is represented by,

\[ k = \frac{2}{2 - \phi - \sqrt{\phi^2 - 4 \phi}} \]  

where \( \phi = c_1 + c_2 \) and \( \phi > 4 \)

By including the constriction factor, the modified velocity and position of each particle can be calculated as using the following formulas:

\[ V_i^{k+1} = k*(w* v_i^k + c_1*r_{\text{rand}}*\text{(pbest}_i - x_i)) + c_2*r_{\text{rand}}*\text{(gbest}_i - x_i) \]  

\[ X_i^{k+1} = x_i + v_i^{k+1} \]

**Application of PSO to economic and emission dispatch problem**

In order to solve a constrained economic and emission dispatch problem, a PSO algorithm was developed to obtain efficiently a high-quality solution within practical power system operation. The PSO algorithm was utilized mainly to determine the optimal lambda and hence power generation of each unit that was submitted to operation at the specific period, thus minimizing the total emission and generation cost.

**Representation of individual**

For an efficient evolutionary method, the representation of chromosome strings of the problem parameter set is important. The proposed approach uses the equal system incremental cost (\( f_{\text{eq}} \)) as individual (particles) of PSO. Each individual within the population represents a candidate solution for solving the emission and economic dispatch problem. The advantage of using system Lambda instead of generator units' output is that, it makes the problem independent of the number of the generator units and also number of iterations for convergence decreases drastically. This is particularly attractive in large-scale systems.

**Evaluation function**

The evaluation function must be defined (it is called fitness function in GA) for evaluating the fitness of each individual in the population. In order to emphasize the “best” chromosome and speed up convergence of the iteration procedure, the evaluation value is normalized into the range between 0 and 1. The evaluation function is adopted as

\[ f = \frac{1}{1 + k \left( \sum_{i=1}^{n} P_i - P_D - P_{\text{loss}} \right) / P_D} \]  

Where, \( k \) is a scaling constant (\( k = 50 \) in this study).

In order to limit the evaluation value of each individual of the population within a feasible range, before estimating the evaluation value of an individual, the generation power output must satisfy the entire \( P_i \) limits.

**Comparison between genetic algorithm and PSO**

Most of evolutionary techniques have the following procedure:

1. Random generation of an initial population.
2. Calculating of a fitness value for each subject. It will directly depend on the distance to the optimum.
3. Reproduction of the population based on fitness values.
4. If requirement are met, then stop. Otherwise go back to 2.

From the procedure, it is understood that PSO shares many common points with GA. Both algorithms start with a group of a randomly generated population, both have fitness values to evaluate the population. Both update the population and search for the optimum with random techniques. Both systems do not guarantee success.

However, PSO does not have genetic operators like crossover and mutation. Particles update themselves with the internal velocity. They also have memory, which is important to the algorithm.
Compared with genetic algorithms (GAs), the information sharing mechanism in PSO is significantly different. In GAs, chromosomes share information with each other. So the whole population moves like a one group towards an optimal area. In PSO, only gbest (or lbest) gives out the information to others. It is a one-way information sharing mechanism. The evolution only looks for the best solution. Compared with GA, all the particles tend to converge to the best solution quickly even in the local version in most cases.

Algorithm

The algorithm for solving the combined emission and economic dispatch problem using PSO method is given below:

1. Read generator data, emission data, P limits, B-coefficients and power demand, population size and number of iterations.
2. Compute the modified price penalty factor \( h_m \) as per steps discussed above.
3. The minimum and maximum limits of Lambda are computed using the incremental cost (\( \lambda_{fe} \)) equation. The particles are generated by selecting a value with a uniform probability over the search space (\( \lambda_{fe}^{\text{min}}, \lambda_{fe}^{\text{max}} \)).
4. The initial velocities of all particles are generated randomly between the velocity limits (\(-V_{max}, V_{max}\)). The maximum velocity is given by
   \[
   V_{max} = \frac{\lambda_{fe}^{\text{max}} - \lambda_{fe}^{\text{min}}}{N_a}
   \]  
   Where, \( N_a \) is number of intervals (10 are taken in this work).
5. Calculate the generation output of all the units for each particle from its \( \lambda_{fe} \) value using equation
   \[
   P_i = \frac{1 - \left( b_i + h_m \beta_i \right)}{\lambda_{fe} - \sum_{j=1}^{n} 2B_gP_j} \cdot \frac{2a_i + 2h_m \alpha_i}{\lambda_{fe} - \sum_{j=1}^{n} 2B_gP_j}
   \]  
   and enforce \( P_i \) limits.
6. Calculate transmission losses using B-coefficient equation.
7. Calculate the evaluation value of each individual in the population using the evaluation function given by eqn.(11). These values are set as \( pbest \) value of the particles.
8. Compare each individual’s evaluation value with its \( pbest \). The best evaluation value among the \( pbest \) is denoted as gbest.
9. Using the \( gbest \) and the individual best (\( pbest \)) of each particle, new velocities are calculated using the eqn. (10a). If a particle violates the velocity limits, set its velocity equal to the limit (i.e. If \( v_i^{k+1} > V_{max} \), then \( v_i^{k+1} = V_{max} \) and if \( v_i^{k+1} < -V_{max} \), then \( v_i^{k+1} = -V_{max} \)).
10. Based on the updated velocities, each particle changes its position according to the following eqn. (10b).
11. With the positions calculate the generation output of all the units using eqn. (13) and enforce the \( P_i \) limits.
12. Calculate transmission losses using B-coefficient equation and compute the error using equation,
   \[
   e = \sum_{i=1}^{n} \left| P_i - P_D - P_{loss} \right|
   \]  
   (14)
13. New evaluation values are calculated for the new positions of the particles. If the new evaluation value is better than the previous \( pbest \) values, the current value is set as \( pbest \). If the best \( pbest \) is better than \( gbest \), the value is set to be \( gbest \).
14. If the number of iterations is not reached maximum or error is greater than pre-specified value (tolerance), go to step 5.
15. With the \( gbest \), compute the optimum values of generation using the eqn. (13) and enforce the \( P_i \) limits.
16. Calculate the total fuel cost, total emission release, emission cost, transmission loss etc. Print the result.

RESULTS

The effectiveness of the proposed method is tested with three generating units. A Binary coded Genetic Algorithm (GA) and proposed PSO method are applied to solve the problem. At each sample system, under the same evaluation function and individual definition, 50 trials were performed using the GA and PSO methods and best result is tabulated.

A reasonable loss coefficients matrix of power system network was employed to draw the transmission line loss and satisfy the transmission capacity constraints. The program is written in MATLAB software package and executed on a P-IV personal computer @1.5 GHz.

Although the PSO method seems to be sensitive to the tuning of some weights or parameters, according to the experiences of many experiments, the following PSO and GA parameters can be used.

GA method

- Number of bits = 12
- Population size = 60
- Number of iterations = 250
- Elitism probability = 0.15
- Cross over probability = 0.7
- Mutation probability = 0.01

The equal system incremental cost (\( \lambda \)) is used as the encoded parameter in the string.
PSO method
- Number of particles = 10,
- Inertia weight parameter \( w \) is set by (8) where \( w_{\text{max}} = 0.9 \) and \( w_{\text{min}} = 0.4 \)
- Learning factor, \( c_1 = 2.02 \) and \( c_2 = 2.02 \)
- Maximum number of iterations = 250

Three-unit system
The generator cost coefficients, emission coefficients and generation limits of three units system are taken from [10]. Transmission loss for this system is calculated using B-coefficient matrix and is given. ELD solution for the three-unit system is solved using evolutionary algorithms such as GA and PSO. Tables 2 and 3 summarize all the results of for various load demands. Comparison of total cost obtained from GA and PSO in combined economic and emission dispatch for load of 700MW is shown in Figure-2.

Result analysis
As seen in tabulated results, the PSO method can obtain lower fuel cost and emission release than the GA method, thus resulting in the higher quality solution. This is, because the PSO method does not perform the selection and crossover operations in evolutionary processes, it can save some computation time compared with the GA method, thus these data are evidence of the superior properties of the PSO method.

CONCLUSIONS
Algorithms have been developed for the determination of the global or near-global optimal solution for the Combined Economic and Emission Dispatch (CEED) problem. The solution algorithms have been tested for test system with three generating units. The PSO approach has demonstrated an ability to provide accurate and feasible solutions within reasonable computation time.

Scope for further work
This paper gives solution of economic dispatch problem for thermal units only. This can be used as a sub-problem for hydrothermal scheduling. It will give complete solution of economic dispatch problem. In this paper it is assumed that the unit commitment is known priori. So unit commitment can be done using PSO technique, and can be integrated to this part.

| Table-1. Cost and emission coefficients of 3- unit system. |
|---|---|---|---|---|---|---|
| Unit | \( a_i \) | \( b_i \) | \( c_i \) | \( a_i \) | \( b_i \) | \( c_i \) |
| 1 | 0.03546 | 38.30553 | 1243.5311 | 0.00683 | -0.54 | 551 |
| 2 | 0.02111 | 36.32782 | 1658.5696 | 0.00461 | -0.51 | 160 |
| 3 | 0.01799 | 38.27041 | 1356.6592 | 0.00461 | -0.51 | 160 |

The Loss Coefficient Matrix of 3- Unit System

\[
B = \begin{bmatrix}
0.000071 & 0.000030 & 0.000025 \\
0.000030 & 0.000069 & 0.000032 \\
0.000025 & 0.000032 & 0.000080
\end{bmatrix}
\]

![Figure-2](image-url) Comparison of total cost obtained from Conventional method, GA and PSO for a 3- unit system (considering only power limits).
Table-2. Combined economic and emission dispatch- 3 unit system (considering only power limits).

<table>
<thead>
<tr>
<th>Gl. No.</th>
<th>Power Demand (MW)</th>
<th>Method</th>
<th>$P_1$ (MW)</th>
<th>$P_2$ (MW)</th>
<th>$P_3$ (MW)</th>
<th>$S_{230}$ (MW)</th>
<th>Fuel Cost (Kg/hr)</th>
<th>Emission Release (Kg/hr)</th>
<th>Emission Penalty Factor (Kg/hr)</th>
<th>Total Cost (Kg/hr)</th>
<th>Execution Time (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>350</td>
<td>GA</td>
<td>155.865</td>
<td>131.522</td>
<td>131.508</td>
<td>5.64555</td>
<td>18551.8</td>
<td>159.118</td>
<td>45.3001</td>
<td>25451.6</td>
<td>0.159</td>
</tr>
<tr>
<td>2</td>
<td>400</td>
<td>GA</td>
<td>189.627</td>
<td>189.627</td>
<td>189.627</td>
<td>7.41524</td>
<td>20400.1</td>
<td>200.056</td>
<td>46.5599</td>
<td>29565.2</td>
<td>0.239</td>
</tr>
<tr>
<td>3</td>
<td>450</td>
<td>GA</td>
<td>130.657</td>
<td>130.657</td>
<td>130.657</td>
<td>9.45775</td>
<td>23142.6</td>
<td>250.399</td>
<td>45.8155</td>
<td>31034.0</td>
<td>0.417</td>
</tr>
<tr>
<td>4</td>
<td>500</td>
<td>GA</td>
<td>120.997</td>
<td>192.082</td>
<td>192.082</td>
<td>11.6924</td>
<td>25499.4</td>
<td>311.272</td>
<td>44.0722</td>
<td>35220.1</td>
<td>0.722</td>
</tr>
<tr>
<td>5</td>
<td>550</td>
<td>GA</td>
<td>145.254</td>
<td>212.217</td>
<td>209.793</td>
<td>14.2164</td>
<td>27905.4</td>
<td>361.238</td>
<td>44.3309</td>
<td>44010.0</td>
<td>1.054</td>
</tr>
<tr>
<td>6</td>
<td>600</td>
<td>GA</td>
<td>145.302</td>
<td>212.784</td>
<td>209.793</td>
<td>14.2151</td>
<td>27904.1</td>
<td>361.216</td>
<td>44.3309</td>
<td>44006.0</td>
<td>1.054</td>
</tr>
<tr>
<td>7</td>
<td>650</td>
<td>GA</td>
<td>155.714</td>
<td>231.895</td>
<td>229.479</td>
<td>17.0039</td>
<td>30373.2</td>
<td>461.352</td>
<td>44.5905</td>
<td>50948.2</td>
<td>2.003</td>
</tr>
<tr>
<td>8</td>
<td>700</td>
<td>GA</td>
<td>153.711</td>
<td>331.859</td>
<td>249.328</td>
<td>16.9907</td>
<td>30368.2</td>
<td>461.207</td>
<td>44.5905</td>
<td>50937.3</td>
<td>2.348</td>
</tr>
<tr>
<td>9</td>
<td>750</td>
<td>GA</td>
<td>145.151</td>
<td>221.625</td>
<td>249.257</td>
<td>20.0477</td>
<td>32688.5</td>
<td>551.599</td>
<td>44.8552</td>
<td>69619.0</td>
<td>4.065</td>
</tr>
</tbody>
</table>

REFERENCES


