We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

The Search for Parameters and Solutions: Applying Genetic Algorithms on Astronomy and Engineering

Annibal Hetem Jr.
Universidade Federal do ABC
Brasil

1. Introduction

Genetic Algorithms (GAs) can help solving a great variety of complex problems, and the characterization of these problems as possible subject for GA is the first step in applying this technique. After some years, we have used this strong tool to solve problems from astronomy and engineering, and both fields demand complex models and simulations.

With the aim of improving previous models and test new ones, we have developed a methodology generate solutions based on GAs. From a first analysis, one must establish the model input and output parameters, and then workout on the inversion of the problem, what we called the inverted model. This concept leads to the final formalism that can be subject to the GA implementation.

After a brief presentation of the main concerns and ideas, it will be described some applications and their results and discussions. Some details on implementation are also given together with the particularities of each model/solution. A special section regarding error bars estimates is also provided. The GA method gives a good quality of fit, but the range of input parameters must be chosen with caution, as unrealistic parameters can be derived.

GAs can also be used to verify if a given model is better than another for solving a problem. Even considering the limitation of the derived parameters, the automatic fitting process provides an interesting tool for the statistical analysis large samples of data and the models considered.

2. Characterization of NP-Complete problems

In this section, the NP-Complete problems are presented as the main targets of GAs. Before starting to project a GA, it is of greatest importance to study and characterize the problem to justify the technique to use.

The early first notion of NP-completeness was proposed by Stephen Cook (1971), in his famous paper *The complexity of theorem proving procedures*. The main ideas presented in this section have their origins in the excelent works of Garey & Johnson (1979) and Papadimitriu (1995).

Deep inside any GA code there is a model of the inverted problem to be solved. This routine works like I don't know what the correct answer is, but I kwon if a candidate to an answer is good or bad. So, the problem to be solved by a GA must have the property that any proposed solution to an instance must be quickly checked for correctness. For one thing, the solution must be concise, with length polynomially bounded by that of the instance.

To formalize the notion of quick checking, we will say that there is a polynomial-time algorithm that takes as input instance and the solution and decides whether or not it is a solution. If a problem demands a nondeterministic polynomial time to be solved, it is said a NP-problem, as defined by complexity theory researchers. It means that a solution to any search problem can be found and verified in polynomial time by nondeterministic algorithm.

2.1 Inverting the problem

The most remarkable characteristic of a NP-complete problem is the lack known algorithms to find its solution. In a P-Problem, any given candidate to solution can be verified quickly for its accuracy or validity. On the other hand, the time required to solve a NP-problem using any currently known search algorithm increases exponentially with the size of the problem grows. As a consequence, one of the principal unsolved problems in computer science today is determining whether or not it is possible to solve these problems quickly, called the P versus NP problem.

Then, suppose one has a problem M to be solved and asks if a GA based program could solve it. The steps to be followed are:

1. To write down formally the set of parameters to be found, something like $S=\{p_1, p_2, p_3, …, p_n\}$, where the p_i set is a representation of the input parameters. Each p_i must be a single number (float or integer), so the S set could be interpreted as a chromosome and each p_i as a gene.
2. To express the problem as a function of the set of parameters: $M=f(S)$, with $M=\{q_1, q_2, q_3, …, q_m\}$, where the q_i set is the representation of the output (desired) parameters.
3. Obtain the inverse problem, or the formalities need to compute $S= g(M) =f^{-1}(M)$.

If the $g(M)$ function can be translated to a writable algorithm, and this algorithm is computable in a finite time, then the $g(M)$ is a P-problem. If the $f(S)$ function cannot be translated to a writable algorithm, or this algorithm is computable only with by verifying all possibilities in the S space, then the $f(S)$ is a NP-problem.

With both answers: the $f(S)$ function is a NP-problem, and its inverse, $g(M)$ is a P-problem, then the problem can be solved by a GA.

3. Applications on astrophysics

Astrophysics is a field of research very rich in NP-complete problems. Many of actual astrophysicists deal with non-linear systems and unstable conditions. In some cases, the comparative data, or the environment in GA jargon, is an image originated in telescopes or instruments placed in deep space. It is common the need for fit models with multi-spectral

data, like radio, infrared, visible and gamma-rays. All these solution constraints lead to an incredible variety of possibilities for using GA tools.

In this section, it will be presented how GAs were used to model protoplanetary discs, an application that involves non-linear radiative-density profile relations. The model combines spectral energy distribution, observed in a wide range of the electromagnetic spectrum, and emissivity behaviour of different dust grain species.

Another interesting application is the use of GAs together with and spectral synthesis in the calculation of abundances and metallicities of T Tauri stars. In this problem, the model is outside the GA code, as one of the conditions imposed is to use a standard, well tested, spectral generator. It is presented how to deal with the challenge of changing a ready to use tool into a NP-complete problem and invert it.

3.1 Using GA to model protoplanetary discs

This subsection is based on the published work *The use of genetic algorithms to model protoplanetary discs* (Hetem & Gregorio-Hetem 2007).

During its formation process, a young star object (YSO) can be surrounded by gas, dust grains and debris, that shall be gravitationally (and also electrostatically) agglomerate in the future solar system bodies. This material receives the energy brought from the star surface and re-irradiates it in other wavelengths. The contribution of this circumstellar matter to the spectral energy distribution (SED) slope is often used to recognize different categories of young YSOs by following an observational classification based on the near-infrared spectral index (Lada & Wilking 1984; Wilking, Lada & Young 1989; André, Ward-Thompson & Barsony 1993). Actually, this classification suggests a scenario for the evolution of YSOs, from Class 0 to Class III, which is well established for TTs.

Here, the adopted model is a flared configuration, according to Dullemond et al. (2001) modelling of a passively irradiated circumstellar disc with an inner hole. We used this model as the P-problem core of a GA based optimization method to estimate the circumstellar parameters.

3.1.1 Presenting the problem

In this subsection we describe the implementation of the GA method for the flared-disc model.

The SED for a given set of parameters is evaluated according to Dullemond et al. (2001) model equations. The disc is composed by three components: the inner rim, the shadowed region, and the flared region with two layers: an illuminated hot layer and an inner cold layer. The disc parameters are: radius, R_D; mass, M_D; inclination, θ; density power law index, p; and inner rim temperature, $T_{\rm rim}$. The stellar parameters are: distance, d; mass, $M_\star$; luminosity, $L_\star$; and temperature, $T_\star$.

The model starts by establishing a vertical boundary irradiated directly by the star, which considers the effect arising from shadowing from the rim, and the variations in scale height as a function of the radius. Figure 1 presents the obtained SED for the star AB Aurigae, as presented in Hetem & Gregorio-Hetem (2007).
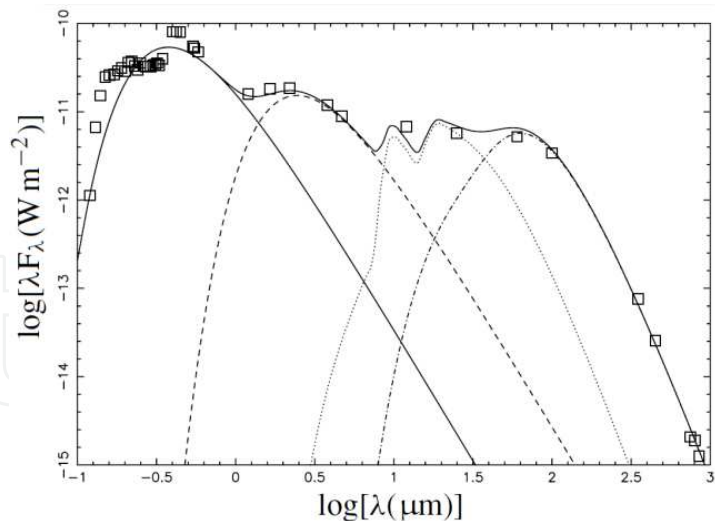
Fig. 1. Results from Dullemond et al. (2001) model applied to the star AB Aurigae. The Synthetic SED is the sum of its components: star emission (continuous thin line); rim emission (dashed line); disc cold layer emission (dot–dashed line); and the disc hot layer emission (dotted line). The observational data in various wavelengths is represented by squares (Hetem & Gregorio-Hetem 2007).

3.1.2 Implementation

The GA code was designed and built to find the best disk parameters, namely $S = \{R_D; M_D; \theta; p; T_{rim}, d; M_\star; L_\star; T_\star\}$, as discussed in subsection 2.1. However, some of these parameters are already known: the stellar parameters d, $L_\star$; and $T_\star$ are adopted from observations and easily found in literature. Essentially, the GA method used implements a χ^2 minimization of the SED fitting provided by the Dullemond et al. (2001) model. The main structures used to manipulate the data are linked lists containing the solutions (parameter set, adaptation level, χ^2_i, and the genetic operator, Φ_i), expressed by

$$M_i = \left\{ \left(R_{Di}, \theta_i, M_{Di}, p_i, T_i \right), \left(\chi_i^2, \Phi_i \right) \right\} \tag{1}$$

where S_i denotes the ith solution, and T_i is the ith Trim. Following Goldberg (1989), the code starts with the construction of the first generation, where all parameters are randomly chosen within an allowed range (for example, $50 \leq RD \leq 1000$ AU). We chose as the number of individuals (parameter sets) in all the generations to be 100. In the following interactions loops, the evaluation function runs the Dullemond et al. (2001) model for each individual, and compares the synthetic SED with the observed data through a χ^2 measure, using the modified expression (Press et al. 1995):

$$\chi_i^2 = \frac{1}{N} \sum_j^N \left(F_j - \varphi_{ij} \right)^2 \tag{2}$$

where F_j is the observed flux at wavelength λ_j, N is the number of observed data points, and $\varphi_{i\,j}$ is the calculated flux for the solution S_i. The smallest χ^2 is assumed to be the *gof*, the goodness-of-fit measure for that generation. The evaluation function is applied to all individuals, and then the judgement procedure sorts the list by increasing χ^2. It also sets one

The Search for Parameters and Solutions:
Applying Genetic Algorithms on Astronomy and Engineering
165

of the genetic operators to the field Φ_i: copy, crossover, mutation or termination. Each Φ is attributed to a fraction of the number of individuals following the values suggested by Koza (1994), Bentley & Corne (2002) and references therein.

With the genetic operators chosen, the next generation is evaluated by applying specific rules according to the genetic operators. The copy operator uses an elitist selection, as the solutions with the smallest χ^2_i are copied to the next generation. For the crossover operator, a random mix of two distinct individuals' genes is built. The mutation operator copies the original individual, except for one of the genes, which is randomly changed. The process loop continues to build new generations until the end condition is reached, as illustrated by the schematic view in figure. 2.
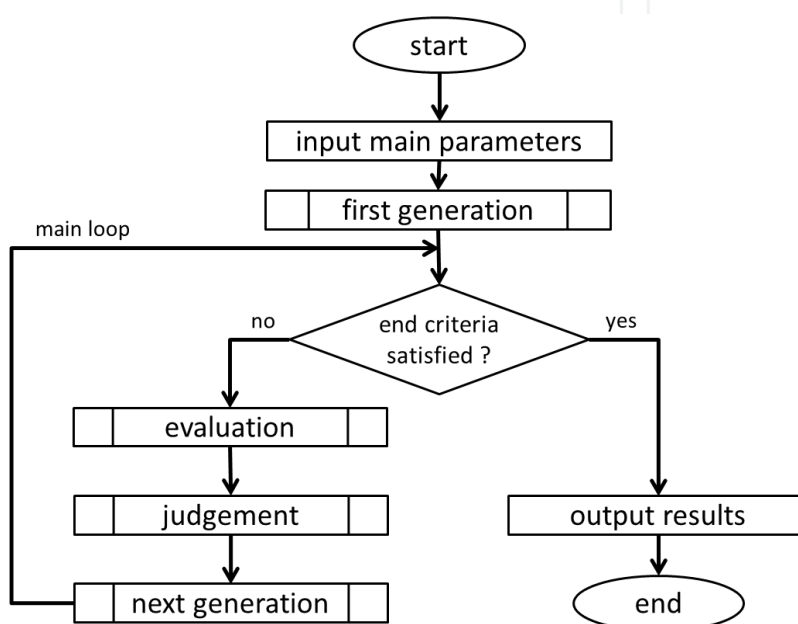

Fig. 2. Main steps of a generic GA (adapted from Hetem & Gregorio-Hetem 2007).

We also can estimate the error bars in the final results by analysing the χ^2 behaviour as a function of the parameter variation. Then one can determine the confidence levels of a given parameter, as suggested by Press et al. (1995). Once the GA end condition has been reached, one can evaluate the inverse of the Hessian matrix $[C] \equiv [\alpha]^{-1}$ whose components are given by

$$\alpha_{ij} = \sum_{k=1}^{N} \left(\frac{\partial y(\lambda_k)}{\partial a_i} \frac{\partial y(\lambda_k)}{\partial a_j} \right) \tag{3}$$

where $\partial y(\lambda_k)/\partial a_i$ is the partial derivative of the SED with respect to parameter a_i at $\lambda = \lambda_k$, and N is the number of observed data points. The main diagonal of C can be used to estimate the error bars on each parameter by $\sigma_i \cong C^{1/2}/N$. We estimated the error bars for the 1σ confidence level and the respective disc parameters for AB Aurigae, resulting in $M_D = 0.1 \pm 0.004 M_\odot$, $R_D = 400 \pm 44$ AU, $\theta = 65 \pm 3^\circ$, and $T_{rim} = 1500 \pm 26$ K, and these results are in agreement with the error-bar estimation provided by the surface contour levels described below. Fig. 3 presents the contour levels of the $gof(M_D, R_D)$ surface calculated for a set of 400

random pairs of disc mass and radius around the parameters for the AB Aurigae model taken from Dominik et al. (2003). The result at the minimum is *gof* ~ 0.046, what means that the error bar estimation converged to a narrow range around the parameter set.
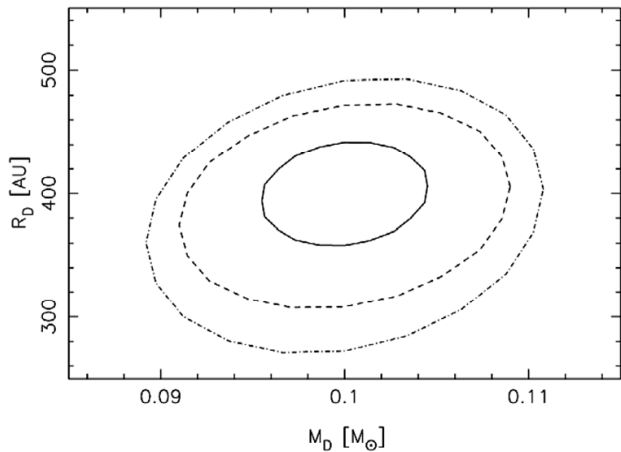

Fig. 3. Contour levels *gof*(M_D, R_D) estimated for AB Aurigae presenting the confidence levels χ^2(68%)= 0.082 (continuous line), χ^2(90%)= 0.15 (dashed) and χ^2(99%)= 0.21 (dot–dashed) (Hetem & Gregorio-Hetem 2007).

We also applied the described GA method to a four other stars, in order to verify the quality of the fitting for objects showing different SED shapes and different levels of infrared excess. Our set was chosen by the slope of their near-infrared SED. The infrared excess in Herbig Be stars is the result of a spherical dusty envelope (van den Ancker et al. 2001), whereas a thick-edge flared disc are characteristic of Herbig Ae. With this in mind, we selected A-type or late-B-type stars from the Pico dos Dias Survey sample (Gregorio-Hetem et al. 1992; Torres et al. 1995; Torres 1998) to apply the GA SED fitting. The results are presented in table 1 together with their corresponding *gofs* (see figure 4).
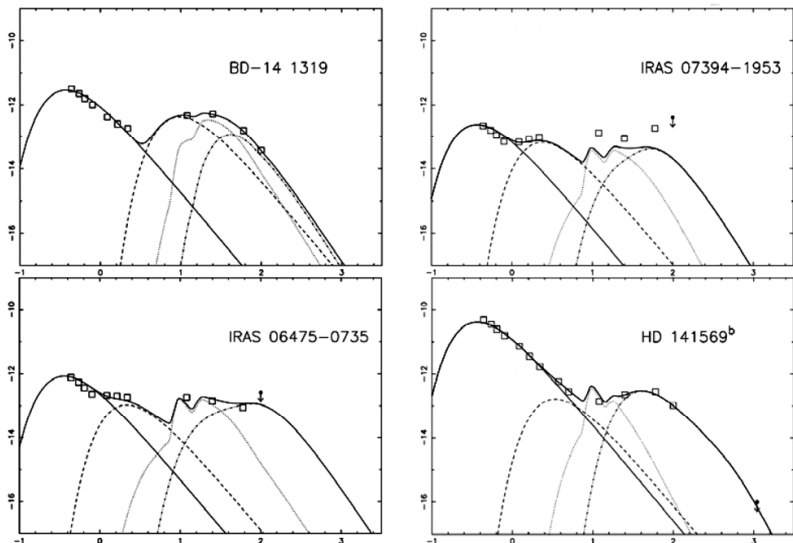

Fig. 4. GA SEDs obtained the stars BD-14 1319, IRAS 07394-1953, IRAS 06475-0735 and HD 141569. The plots are given as *log*[λF_λ(Wm^{-2})] versus *log*[λ(μm)] (Hetem & Gregorio-Hetem 2007).

PDS	Name	$M_\star$ (M$_\odot$)	R_D (AU)	M_D (M$_\odot$)	T_{rim} (K)	θ (o)	p	gof
398	HD 141569	2.4	13	0.06	1085	0.6	-2.0	0.006
022	BD−14 1319	2.8	690	0.003	380	40	-10.	0.006
130	IRAS 06475−0735	2.0	309	0.20	1705	53	-1.5	0.016
257	IRAS 07394−1953	2.0	859	0.64	1838	47	-2.0	0.098

Table 1. Obtained parameters for the chosen stars (Hetem & Gregorio-Hetem 2007).

3.2 Abundances and Metallicities of young stars via Spectral Synthesis

This subsection is based on the published work The use of Genetic Algorithms and Spectral Synthesis in the Calculation of Abundances and Metallicities of T Tauri stars (Hetem & Gregorio-Hetem 2009).

In the previous subsection, we presented a method that uses a calculation technique based on GA aiming to optimize the parameters estimation of protoplanetary disks of T Tauri stars. Inspired by the success of that application, which gives accurate and efficient calculations, we decided to develop a similar method to determine atomic stellar abundances.

3.2.1 Artificial spectra as a measurement tool

In astrophysics, the absorption spectra are obtained and employed as an analytical chemistry tool to determine the presence of atoms and ions in stellar atmospheres and, if possible, to quantify the amount of the atoms present. In stellar atmospheres, each element produces a number of spectrum absorption lines, at wavelengths which can be measured with extreme accuracy when compared to spectra emission tables provided by laboratory experiments.

The presence of a given element in the star atmosphere can be verified (and measured) by looking for its absorption lines at the correct wavelength. The hydrogen is present in all stars by its Balmer absorption lines, and is often used to calibrate the measurements. An example of a high-resolution spectrum is presented in figure 5.

The way astrophysics use to calculate the abundances of atoms in stars follows the steps:

1. Obtain the star spectrum in a given range (or ranges) of wavelength, where the lines of the elements in study should be;
2. Generate an artificial spectrum, considering the lines whose origin are the desired elements and the known physics of absorption line production;
3. Compare the artificial and observed spectra. Here a simple χ^2 test is enough to compute a general comparison index;
4. Use a GA methodology to optimize the artificial spectrum in order to minimize the differences with the observed spectrum (the inverted problem, subsection 2.1);
5. Once the optimization methodology reaches its goals, consider the elemental parameters (density, temperature, ionization, etc) as the measures of the elements in the stellar atmospheres.

3.2.2 Inverting the problem

From our discussion on section 2, one can see that generating a synthetic spectrum is a P-problem, as the result is obtained from a set of parameters, and no more computing is need. The generation time is obviously finite, and there are a number of very efficient software tools that do that. The only care to be taken is to assure that the artificial spectrum has the same wavelength resolution of the observed spectrum, in order to simplify the future comparison.

The above mentioned step 4, a methodology to optimize the artificial spectrum, is the trick point. If one wants to use GA so solve the abundances problem, it is necessary to invert the P-problem, that is, it is necessary to use the artificial spectrum generation tool as an external routine of a bigger and more complex algorithm. The algorithm used to this task is presented in figure 6.
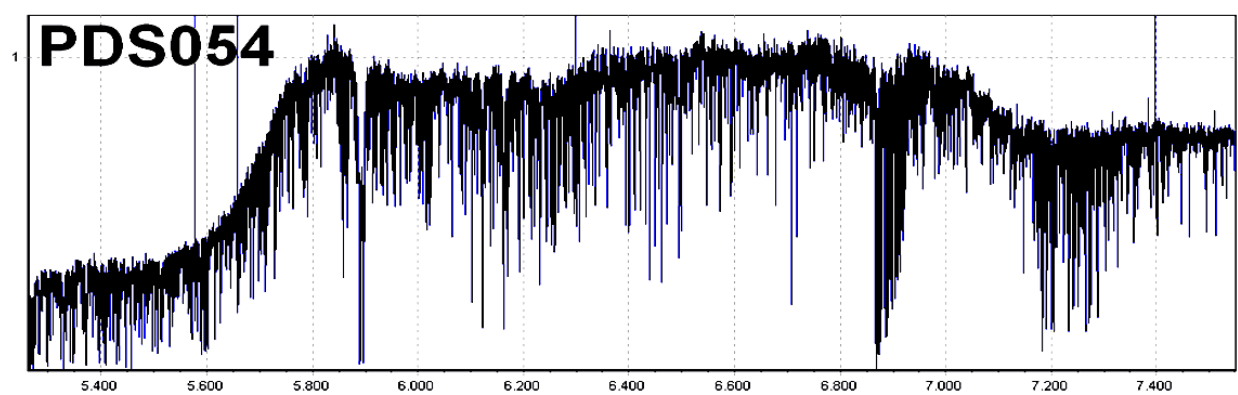

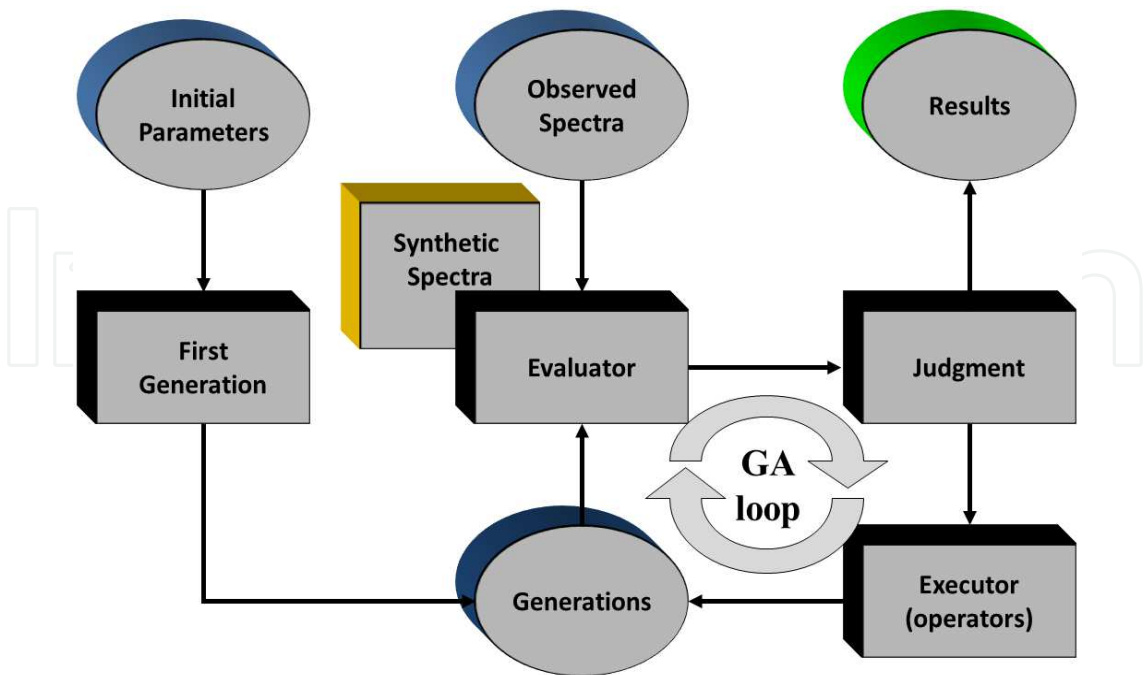Fig. 5. FEROS spectrum for star PDS054 (Rojas et al. 2008).

Fig. 6. Main blocks of a GA code to fit multi-band spectra of T Tauri stars (adapted from Hetem & Gregorio-Hetem 2009).

Establishing the abundances of each element as the parameters to be found, one individual in the GA terminology is the set of all elemental abundances added to some atmospheric parameters. The initial parameter set is used to build the first generation with 100 individuals. The evaluator routine creates a synthetic spectrum whose entries are the genetic data in each chromosome. This task is performed by calling the elected spectral tool.

There are a number of very efficient software tools that can be chosen. In our application, the abundances of chemical species are determined by using the spectral synthesis software SPECTRUM provided by Corbally (Gray & Corbally 1994) and the atmosphere model software ATLAS9 from Kurucz (1993).

3.2.3 Results

In this section we present the results of the GA method for three stars, whose high-resolution spectra were obtained at *European Southern Observatory* (ESO) in La Silla, Chile, with the *Fibber Extended Range Optical Spectrograph* (FEROS) at the 1.52m telescope. The stellar parameters (effective temperature and gravity) were calculated by excitation and ionization equilibrium of iron absorption lines (Rojas et al. 2008). The atomic and molecular line data were mainly from the National Institute of Standards and Technology[1] and the Kurucz site[2]. The solar atomic abundances are from Grevesse & Sauval (1998), and the hyperfine structure constants were taken from Dembczyński et al. (1979) and Luc & Gerstenkorn (1972). The atmosphere models where obtained from the Kurucz library. Specific atmosphere models were calculated through a GNU-Linux porting of the ATLAS9 program (Kurucz 1993).

The method performs a multi-range fitting of specific regions of the observed spectrum, looking for best fit. The demands and commands to SPECTRUM are only those for generating the specific regions of interest, but the χ^2 comparing index is evaluated over all wavelength ranges. Figures 7 and 8 present the results for some stars on chosen lines.

The metallicities and abundances found for the stars are compatible with those previously obtained for this particular sample. These preliminary results, achieved by using the GA technique, indicate the efficiency of the method. In the future, we intend to use the method in a larger sample of T Tauri stars.

4. Applications on Rocket Engine engineering

This section presents two solutions in applying GAs in the aerospace area, both concerning the fuel pumping in liquid propellant rocket engines. There are many choices to be done in the design of a high performance fuel pump, being one of them the type of pump.

Two different types of pumps were modelled: the Harrington pumps and the turbo pumps. Both present a complex design methodology, which includes: tabled functions interpolations, numerical integrals and constructive material choices.

---

[1] http://physics.nist.gov/PhysRefData/ASD/index.html
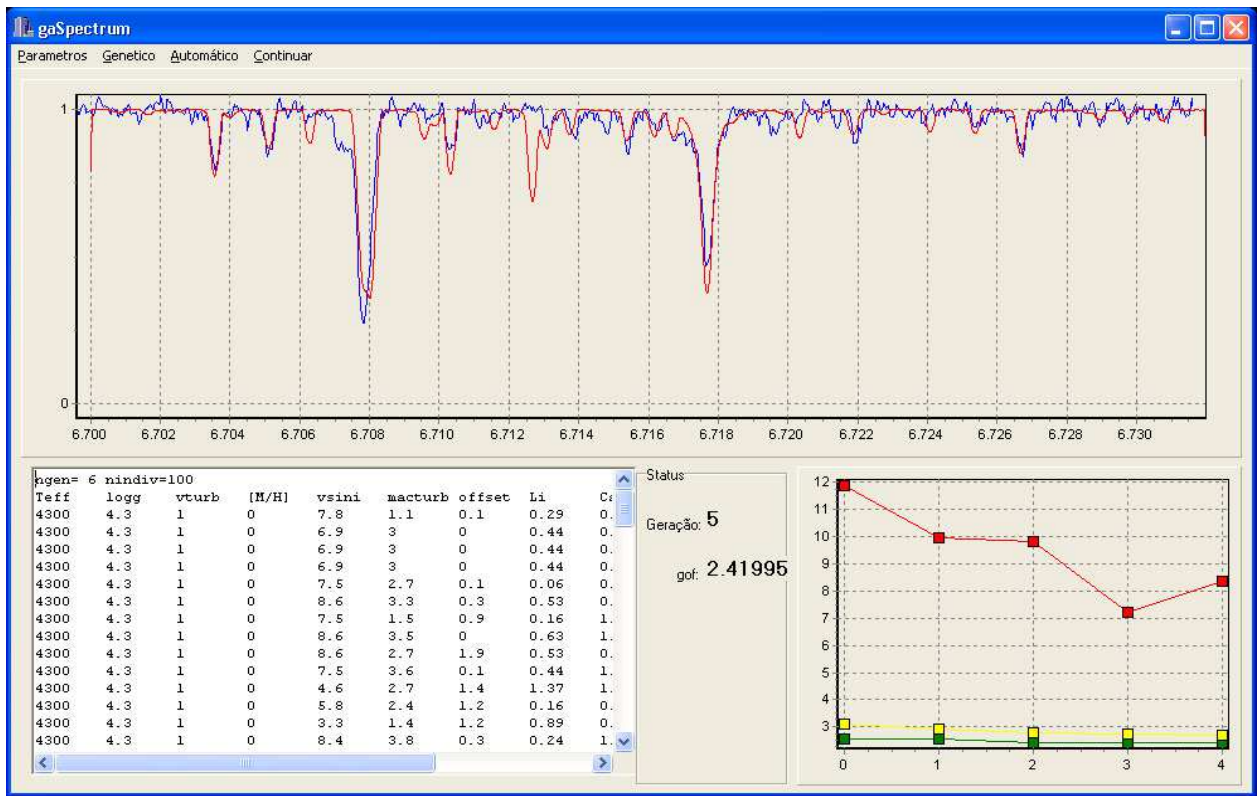[2] http://kurucz.harvard.edu 4 and http://wwwuser.oat.ts.astro.it/castelli/

Fig. 7. Main screen of the program GASpectrum after five generations. The upper panel presents the spectra: the blue line represents the observed spectrum and the red line represents the best individual spectrum (adapted from Hetem & Gregorio-Hetem 2009).
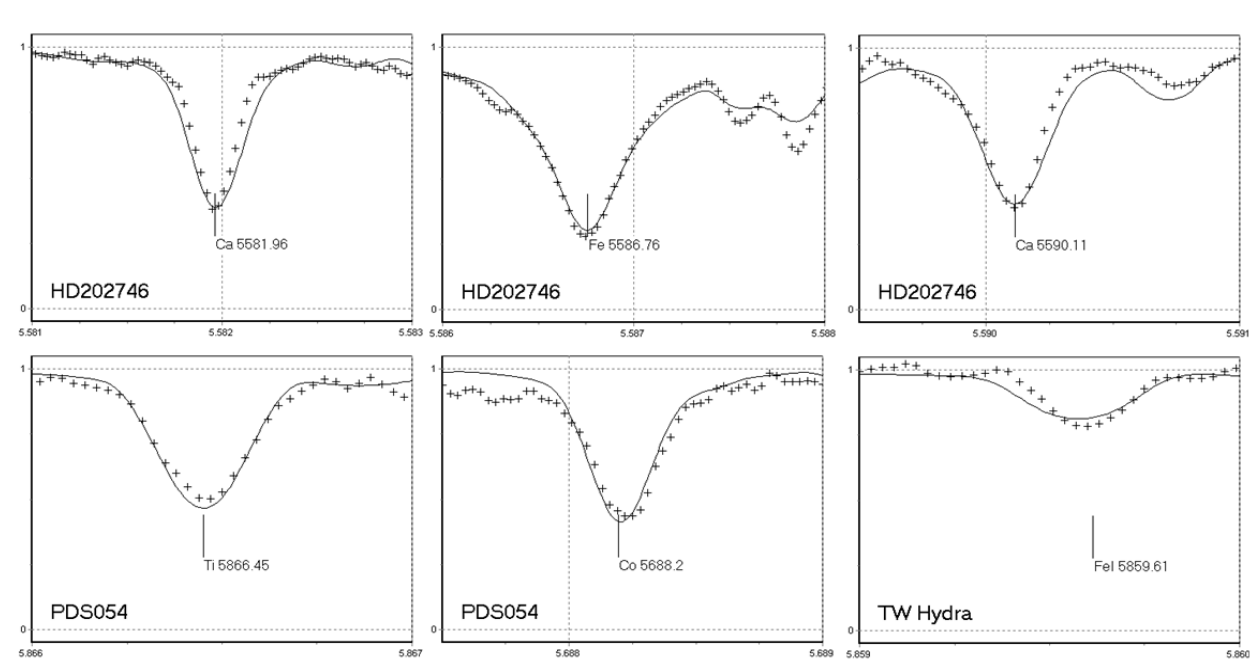

Fig. 8. Main results for stars HD202746, PDS054 and TW Hydra, on calcium, iron, titanium and cobalt lines (adapted from Hetem & Gregorio-Hetem 2009).

4.1 Using GA to parameterize the design of Harrington pumps

This subsection is based on the published work *Artificial Intelligence Parametrization of Harrington Pumps* (Caetano & Hetem 2011).

Since the beginning of liquid engine spacecraft history, the choices on pumping were the turbo pumps (Neufeld 1995). However, turbo pumps present many difficulties to design and to achieve their optimum performance. Good and experienced designers can project specialized turbo pumps that can deliver 70-90% efficiency, but figures less than half that are not uncommon. Low efficiency may be acceptable in some applications, but in rocketry this is a severe problem. Common problems include: 1) excessive flow from the high pressure rim back to the low pressure inlet along the gap between the casing of the pump and the rotor; 2) excessive recirculation of the fluid at inlet; 3) excessive vortexing of the fluid as it leaves the casing of the pump; 4) damaging cavitation to impeller blade surfaces in low pressure zones; and 5) critical shaping of the rotor itself is hardly precise (see the many examples and demonstrations presented by Dixon & Hall (2010) for a better understanding of these concerns).

On the other end, the options are the pressurized tanks. In this choice, the fuel and oxidizer reservoir are filled charged with a high pressure gas (helium or nitrogen) that pushes the fluid to the thrust chamber. So, it is easy to see that the tank output fuel pressure drops as the rocket engine consumes its content. As an option, the designer can increase the inside pressure, but this came also with a high cost in material (due to tank thickness) and instability. Actually, pressurized propellant tanks are used on small rockets like the last stages on space missions.

As an elegant intermediate solution between these two extremes, Harrington (2003) presented a design fills the gap between the pressure fed and the turbo pumps. This solution also has the advantage of lowering the costs of a rocket project, keeping low weight and without the high complexity of a turbo pump, whose operation, theoretical concerns and constructive details are explained in next section.

4.1.1 Pump description and operation

The construction consists of two chambers (B1 and B2 on figure 9) and a set of 8 valves. The chambers are connected to the main tank (Mt) through valves k3 and k4. These chambers also deliver propellant to the combustion chamber (CB) through valves k5 and k6. There is a high pressure gas generator (Hp) that is connected to the chambers through valves k1 and k2. Valves k7 and k8 serve as ventilation for the chambers.

The pumps work alternating two states. In state 1, B1 is being filled by Mt and B2 is feeding the combustion chamber; and in state 2 their role is inverted, say B2 is being filled by Mt and B1 is feeding the combustion chamber. The state change is done by opening and closing the valves, as presented in figure 9 and table 2. The opening and closing of the valves is controlled by a small processor.

4.1.2 The model: Pump constructive details

Designing a Harrington pump is simple, but the optimization process is not (as expected: a P-problem and a NP-problem respectively). A pump with a small chamber must be filled

and vented quickly, with minimal head loss through the gas and liquid valves and plumbing. Making the pump cycle as fast as possible would make it lightweight, but higher flow velocities cause problems (Harrington 2003).
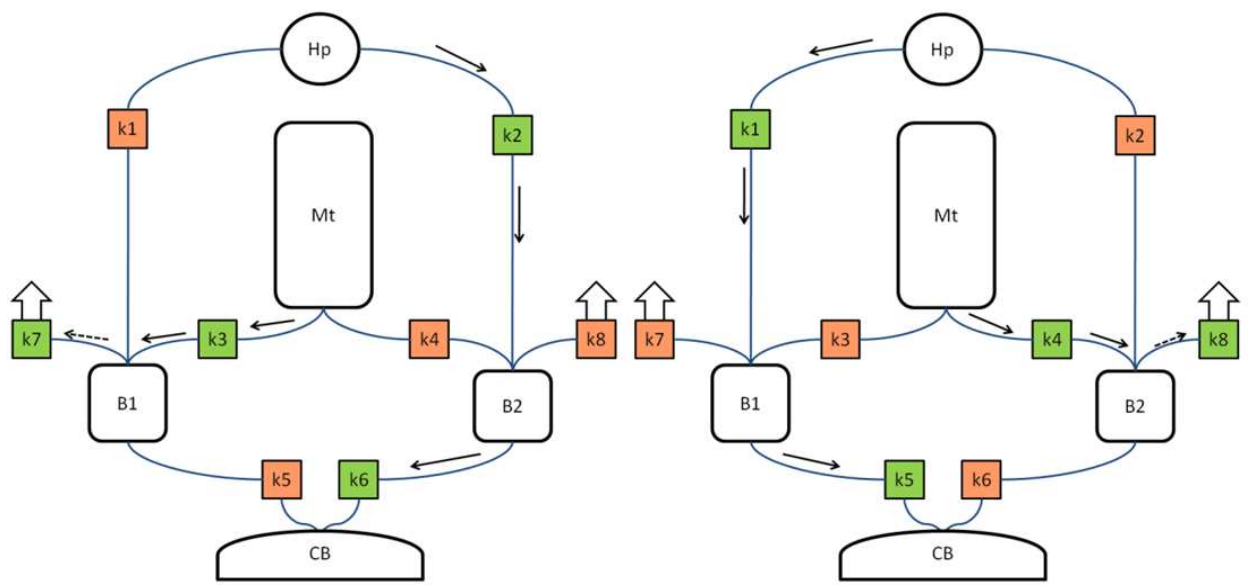
Fig. 9. Schematic view and operation of a Harrington pump, with its chambers (B1 and B2) and valves (k1-8). The main rocket fuel tank is represented by Mt whereas Hp represents a high pressure gas generator. The two states are presented. Left: B1 is being filled by Mt while B2 is feeding the combustion chamber. Right: B1 is feeding the combustion chamber while B2 is being filled by Mt. the arrows indicate the flow. (Caetano & Hetem 2011).

valve	state 1	state 2
k1	closed	open
k2	open	closed
k3	open	closed
k4	closed	open
k5	closed	open
k6	open	closed
k7	open	closed
k8	closed	open

Table 2. Derived model parameters for the sample (Caetano & Hetem 2011).

The choice of pump tanks material plays an important role, as its mass density and stress coefficients are the main keys in the pump design. The main tank pressure (about 300 kPa) and the area of the inlet valves set up the limits for the maximum inflow rate. If the inflow velocity is increased this can cause the propellant to be aerated, what is not desirable for the proper working of the engine. The extra volume of pressurized gas in the pump chamber should be small to minimize gas usage, but if it is too small, there will be a loss of propellant through the vent.

The primary parameters for the calculations are the state changing cycle, t_{cy}, the volume flow determined by the rocket engine needs, Q, the specific impulse of the propellants, I_{sp}, at the fuel pressure, P_f, the fuel mass density, ρ_f, the thrust, T, and the material properties: the mass density, ρ_c, and stress coefficient, σ_c. From these parameters, considering the pump chambers are spherical, one can instantly obtain the diameter of one chamber:

$$D_c = \sqrt[3]{6 \frac{\int_0^{t_{cy}} Q dt}{\pi}} , \tag{4}$$

where the integral results in the chamber volume, and for the simplest case of steady flow, it resumes to $V_c = Q.t_{cy}$. Knowing the diameter and applying the stress formulae from Young (1989), the chamber walls thickness can be obtained by

$$t_w = \frac{P_f D_c}{\sigma_c} , \tag{5}$$

and the total chamber mass by

$$M_c = \pi t_w D_c^2 \rho_c . \tag{6}$$

To obtain the thrust, one can apply the momentum equation for the case of ideal expansion, and:

$$T = g Q I_{sp} \rho_f , \tag{7}$$

where g represents the gravity acceleration.

Manipulation of these expressions and an estimative of the relative weight of the valves and other accessories lead to expression 7 from Harrington (2003), the pump thrust to weight ratio:

$$\frac{T}{W} = 0.43 \frac{g I_{sp} \rho_f}{P_f T_{cy}} \frac{\sigma_c}{\rho_c} , \tag{8}$$

that is to be optimized. The total pump mass is $M_p = 1.56\, M_c$, and the mass flow can be easily obtained by $\dot{m} = Q \rho_f$. The expressions (4)-(8) were coded in a program to test the feasibility of this set of equations as a model. Table 3 presents the results obtained for typical parameter values. These results are in agreement with rocket engine pump literature (Griffinand & French 1991; Sutton 1986).

Entry parameters			Model results		
t_{cy}	5	s	V_c	0,016667	m³
Q	200	l/min	D_c	31,69203	cm
I_{sp}	285	s	t_w	0,090549	cm
ρ_f[1]	935	kg/m³	M_c	0,8	kg
P_f	4	Mpa	M_p	1,248	kg
T	8800	N	T	8704,85	N
σ_c[2]	350	MPa	T/W	8718,8	
ρ_c[2]	2,8	g/cm³	$\dot{m}$	3,116667	kg/s

[1] Propellant mixture: LOX/RP-1
[2] 2219 Aluminum alloy

Table 3. Test values for the pump model and results.

4.1.3 GA optimization method

Here we describe de Genetic Algorithm (GA) optimization method and the formalism applied to code the problem to its needs.

The pump parameters we want to find are a subset of those described as primary parameters: the state changing cycle, t_{cy}, the fuel pressure, P_f, the fuel mass density, ρ_f, and the material properties: the mass density, ρ_c, and stress coefficient, σ_c. These are the GA free parameters, formally

$$\Lambda = \{t_{cy}, P_f, \rho_c, \sigma_c\} , \qquad (9)$$

known as the parameter set. The technique used to work with the material parameters, ρ_c and σ_c, are explained in sub-section 4.1.4.

The obtained pump must deliver a desired mass rate, , of a given propellant, ρ_f, and must be made of a given material, ρ_c and σ_c. Some variables are project dependent, like the volume flow, Q, the specific impulse of the propellants, I_{sp}, at the fuel pressure, and the thrust, T. These three parameters are those the rocket engine designer should define to specify the pump he needs. Differently from the first parameters described on the above paragraph, these values cannot be altered by the algorithm, and can be included in another group, the constant set:

$$\Psi = \{Q, I_{sp}, T\} . \qquad (10)$$

Another group of variables is need: the result set. These are the values that are obtained by running the model code:

$$\Gamma = \{V_c, D_c, t_w, M_c, M_p, T, T/W, \dot{m}\} . \qquad (11)$$

To satisfy the GA formalism, one must write down the model that describes the necessary transformations to obtain Γ from Ψ and Λ, or $\Lambda = f(\Psi, \Gamma)$.

Now we explain how the GA method was implemented in the Harrington pump model described above. We first clarify the GA nomenclature in the field of pump design. A *parameter* (e.g. volume flow) corresponds to the concept of a 'gene', and a change in a parameter is a 'mutation'. A parameter set that yields a possible solution corresponds to a 'chromosome', our Λ. An 'individual' is a solution that is composed of one parameter set and two additional GA control variables. One of these variables is χ^2, which refers to the 'adaptation' level. The other control variable is Φ, the genetic operator. The term 'generation' means 'all the individuals' (or all the solutions) present in a given iteration.

The code uses the parameters described in (9), namely $\Lambda = \{t_{cy}, P_f, \rho_c, \sigma_c\}$. Essentially, the GA method presented herein implements a χ^2 minimization of the comparison between the desired results $\Gamma_0 = \{V_c, D_c, t_w, M_c, M_p, T, T/W, \dot{m}\}$, and the results obtained by the application of expressions (4) to (8), the model results. There are three main advantages of using a GA for this task: (i) the GA method potentially browses the whole permitted parameter space, better avoiding the 'traps' of local minima; (ii) the method is not affected by changes in the model; (iii) the GA implementation does not need to compute the derivatives of χ^2 (such as $\partial \chi^2 / \partial P_f$, for example) required by the usual methods. This fact simplifies the code and minimizes computer errors caused by gradient calculations.

The main structures used to manipulate the data are linked lists containing the solutions (parameter set, adaptation level, χ_i^2, and the genetic operator, Φ_i, expressed by $S_i = \{\Psi, \Lambda_i, \Gamma_i, (\chi_i^2, \Phi_i)\}$, where S_i denotes the ith solution. Following Goldberg (1989) and Hetem & Gregorio-Hetem (2007), the code starts with the construction of the first generation, where all parameters are randomly chosen within an allowed range (for example, 15 cm < D_c < 30 cm). Here, the number of parameter sets in the first generation is assumed to be 100. In the next step, the evaluation function runs the model for each solution, and compares the synthetic Γ_i with the desired data, Γ_0, to find χ^2, using a modified expression given by Press et al. (1995):

$$\chi_i^2 = \frac{1}{n_p} \sum_{j=1}^{n_p} \left(\frac{\Gamma_{0j} - \Gamma_{ij}}{\Gamma_{0j}} \right)^2 , \qquad (12)$$

where n_p is the number of values in the result set, Γ_{0j}, is the desired value on position j (e.g. $\Gamma_{01} = V_c$), and Γ_{ij} is the calculated value for the solution S_i. The smallest χ^2 corresponds to the goodness-of-fit, or simply *gof*. The *gof* values express how each individual is adapted, or how close each solution is, to the best solution (Bentley & Corne 2002). For the value of *T/W*, which we want to optimize, it is enough to establish a corresponding to Γ_{0j} very high.

A judgment function then determines the genetic operator Φ to be applied to a solution. Its values can be 'copy': the individual remains the same in the next generation; 'crossover': the individual is elected to change a number of genes (parameters) with another individual, creating a new one; 'mutation': one of its genes is randomly changed; or 'termination': none of the genes continue to subsequent generations. The chosen action is expressed by the Φ_i variable, associated with each individual. The next step is to evolve the current generation (k) to the next ($k + 1$) one, which is done through a multi-dimensional function β that considers the solutions and the genetic operators. Formally,

$$[S_1, S_2, \ldots, S_N]_{k+1} = \beta \big[(S_1, \Phi_1), (S_2, \Phi_2), \ldots, (S_N, \Phi_N) \big]_k . \qquad (13)$$

As soon as a new generation is ready, the evaluation function is reapplied, and the algorithm repeats the described actions until an end-of-loop condition is reached. The end condition can be based on the number of iterations or the quality (a low level for the χ_i^2 values).

4.1.4 The choice of chamber constructive material

The main material properties, the mass density, ρ_c, and stress coefficient, σ_c, can also be chosen by the GA. Instead of working directly with these parameters, it was created a material parameter, K_c, an integer that points to a density-stress database. So, our new parameter set becomes

$$\Lambda = \{ t_{cy}, P_f, \rho_c(K_c), \sigma_c(K_c) \} , \qquad (14)$$

or simply

$$\Lambda = \{ t_{cy}, P_f, K_c \} . \qquad (15)$$

As K_c is a discrete value, it was needed to build special routines to manipulate the genes in the first generation and in mutation events.

4.1.5 Results and conclusion

Table 4 presents the main results for a GA run of 20 generations. The values are in agreement with the expected for the pump. The material chosen for the chambers was cooper 99.9%. A typical running with about 100 generation is achieved in ~5 seconds in a simple laptop computer.

t_{cy}	8.2	s	Vc	0.00393786	m³
Q	200	l/min	Dc	0.195924	cm
I_{sp}	285	s	tw	0,089	cm
ρ_f [1]	935	kg/m³	Mc	0.957973	kg
P_f	4	Mpa	Mp	1.49444	kg
σ_c [2]	350	MPa	T/W	843.227	
ρ_c [2]	2,8	g/cm³	$\dot{m}$	0.448098	kg/s

[1] Propellant mixture: LOX/RP-1
2 Copper 99.9% Cu

Table 4. GA result values for the pump model.

The GA proved to be efficient, and due to the method itself being independent of model complexity, it certainly can be used in future implementations of pump design. Future

The Search for Parameters and Solutions:
Applying Genetic Algorithms on Astronomy and Engineering
177

evolutions and increasing complexity of the model, like thermal transfer and realistic valves, can benefit of GA robustness and reliability.

The next step in this work is to enhance the model with more realistic and specific trends. It is expected to incorporate non-linear functions, differential equations and integrals. Also tabled functions are not far from what can be found in a pump project, with its intrinsic interpolations. The overall problem of finding parameters for a pump design can easily turn to a NP-Problem, that is a problem that is very difficult to find a solution, but, once one has a candidate to solution it is easy to verify if it is a good solution.

4.2 Using GA to parameterize the design of turbo pumps to be used in rocket engines

This subsection is based on the published work Parametric Design of Rocket Engine Turbo pumps with Genetic Algorithms (Burian et al. 2011).

Turbo pumping in high-thrust, long-duration liquid propellant rocket engine applications, generally results in lower weights and higher performance when compared to pressurized gas feed systems. Turbo pump feed systems require only relatively low pump-inlet pressures, and thus propellant-tank pressures, while the major portion of the pressure required at the thrust chamber inlets is supplied by the pumps, saving considerable vehicle weight. As stated by Huzel & Huang (1967) the best performing turbo pump system is defined as that which affords the heaviest payload for a vehicle with a given thrust level, range or velocity increment: gross stage take-off weight; and thrust chamber specific impulse (based on propellant combination, mixture ratio, and chamber operating efficiency).

The particular arrangement or geometry of the major turbo pump components is related to their selection process (Logan & Roy 2003). Some complex designs, like the SSME-Space Shuttle Main Engine, have a multiple stage pump, but most propellant pumps have a single-stage main impeller. Eventually, one or more design limits are reached which requires more iteration, each with a new changed parameter or approach. For a better example, see table 5 which presents some data from the V2 (II world war German missile) alcohol pump.

Parameter	value
impeller diameter	34 cm
rotation	5000 rpm
performance	265 kW
delivery	50 kg/s
delivery pressure	25 atm

Table 5. Parameters from the alcohol V2 pump, adapted from Sutton & Biblarz (2001).

This subsection considers the development of a software tool based on GA to assist the determination of the excellent parameters for the configuration of turbo pumps in engines

for liquid propellant rockets. We present the first version, which considers the calculation of the main parameters of a compressor stage.

4.2.1 The model

The pump compressor model used in this work is based on chapter 10 of Sutton & Biblarz (2001). This model provides a coherent basis for the modeling, and is sufficiently complex to be used as a valid test on the further parameter optimizing step.

The pump parameters we want to find are: the inlet compressor diameter, d1, the compressor outlet diameter, d_2, the fluid input velocity, v_1, the suction specific speed, S, the shaft cross section, A_{S1}, the pressure in the main tank, P_t, the total fluid friction (viscosity included) due to flow through the pipes, valves, etc, P_f, the pressure due to the tank elevation from the pump inlet, P_e. In particular, this last parameter leads to project insights concerning the pump position inside the rocket. These are the GA free parameters, formally $\Lambda = \{d_1, d_2, v_1, S, d_{S1}, P_t, P_f, P_e\}$, known as the parameter set. The obtained compressor must deliver a desired mass rate, $\dot{m}$, and, from an input pressure P_1, generate a flow with an output pressure P_2. Some constants shall be considered, like the fluid mass density, ρ, and the external gravity, g_0. We assumed as fluid the ethanol (C_2H_6OH) due to its green properties and green results. These three parameters are those the rocket engine designer should define to specify the compressor he needs. Differently from the first eight parameters described on the above paragraph, these values cannot be altered by the algorithm, and can be included in another group, the result set $\Gamma = \{\dot{m}, P_1, P_2\}$.

To satisfy the GA formalism, one must write down the model, or the formalism that describes the necessary transformations to obtain Γ from Λ, or $\Gamma = f(\Lambda)$. One can obtain these expressions following Sutton & Biblarz (2001) model and converting their expressions. First, the pressures should be converted to heads, or the height necessary to the fluid to cause a given pressure, so we define H_t, H_e and H_f, the tank head, the elevation head and the friction head, respectively, that can be obtained by

$$P_t = \int_{H_t} g_0 \rho \, dh \,, \tag{16}$$

$$P_e = \int_{H_e} g_0 \rho \, dh \,, \tag{17}$$

and

$$P_f = \int_{H_f} g_0 \rho \, dh \,. \tag{18}$$

The effective area of the inlet is given by

$$A_{1eff} = \frac{1}{4} \frac{d_1^2}{\pi} - A_{S1} \,, \tag{19}$$

which determines the volume flow

$$Q = \iint\limits_{A_{1\,eff}} v_1 dA \; . \tag{20}$$

Then, the absolute positive head can be obtained by

$$H_1 = H_t + H_e - H_f \tag{21}$$

and the net positive suction head or available suction head above vapor pressure can be obtained by

$$H_s = H_t + H_e - H_f - H_v \; , \tag{22}$$

where H_v is the combustible vapor pressure. The required suction head will be taken as 80% of the available suction head in order to provide a margin of safety for cavitation, or $H_{SR}=0.8H_1$. To avoid pump cavitation, H_s has to be higher than H_{SR}. If additional head is required by the pump, the propellant may have to be pressurized by external means, such as by the addition of another pump in series (a booster pump) or by gas pressurization of the propellant tanks. A small value of H_{SR} is desirable because it may permit a reduction of the requirements for tank pressurization and, therefore, a lower inert tank mass.

The shaft speed is given by

$$N_{rpm} = \frac{SH_{SR}^{\phi}}{u_{SI}\sqrt{Q}} \; , \tag{22}$$

where $\varphi=3/4$ and $u_{SI}=17.827459$ are constants. u_{SI} is necessary due to SI converstions (see Sutton & Biblarz 2001, eq. 10-7). This last expression allows us to obtain $N_{rad/s}$, the shaft speed in radians per second. The impeller vane tip speed is given by

$$u = \frac{1}{2}d_2 N_{rad/s} \; . \tag{23}$$

With u, we can evaluate the head delivered by the pump

$$\Delta H = \frac{u^2}{\psi g_0} \; , \tag{24}$$

where ψ has values between 0.90 and 1.10 for different designs. As for many pumps, $\psi = 1.0$, we adopt this value.

At this point, we are able to obtain all the final results, $\Gamma = \{\dot{m}, P_1, P_2\}$:

$$P_1 = H_1 g_0 \rho \; , \tag{25}$$

$$P_2 = (\Delta H + H_1) g_0 \rho \; , \tag{26}$$

and

$$\dot{m} = \rho Q \; . \tag{27}$$

It is also interesting to evaluate the shaft specific speed

$$N_s = \frac{u_{SI}\sqrt{Q}}{H_{SR}^{\phi}} \, , \tag{28}$$

which, with the aid of table 10-2 of Sutton & Biblarz (2001), defines the pump and impeller type.

4.2.2 Results and conclusion

We built a computer code to optimize equations in the same way it was done to the Harrington pumps (see subsection 4.1). The resulting parameters obtained from the GA code where in good agreement with what is expected for this kind of project. Some comparisons between GA results and correct results are presented in table 6.

		$\dot{m}$ (kg/s)	P$_1$ (Pa)	P$_2$ (Pa)	mean error (%)
Correct answer		226,8	342669	6816870	
generations	10	228,1	342345	6816450	0,22
	20	227,5	342360	6816440	0,13
	50	227,1	342601	6816890	0,05
	100	226,9	342670	6816880	0,01

Table 6. Comparison between obtained results (GA) and correct answer (Γ_0) for an ethanol compressor.

Evidently, for the simple definitions presented for this model, one does not need a sophisticated method as described to obtain a good result. But, as all designers know very well, there are no simple projects, especially concerning rocket engine pumps. The next step in this work is to enhance the model with more realistic and specific trends. It is expected to incorporate non-linear functions, differential equations and integrals. Also tabled functions are not far from what can be found in a pump project, with its intrinsic interpolations. The overall problem of finding parameters for a pump design can easily turn to a NP-Problem, that is a problem that is very difficult to find a solution, but, once one has a candidate to solution it is easy to verify if it is a good solution. Again, the GA proved to be efficient, and due to the method itself being independent of model complexity, it certainly can be used in future implementations. Future evolutions and increasing complexity of the model can benefit of GA robustness and reliability.

5. Applications on energy distribution

The application described in this section solves the problem of allocation of protective devices in electric power distribution plants. For a given power plant distribution, it is necessary to choose in which points one must place equipment for the net protection, or not.

This problem is entirely based on discrete elements – there are no floating point parameters. So, the main discussion here is how to build a chromosome syntax that can be used under the GA rules, and still be meaningful for the model. Besides, as the problem is fully discretized, there are high probabilities of finding different solutions that are equally evaluated in their adaptation function. This leads to new enhancements in the model to better evaluate the solutions, enhancing the separation between different individuals.

5.1 Using GA in the allocation of electric power protective devices

This subsection is based on the published work Automatic Allocation of Electric Power Distribution Protective Devices (Burian et al. 2010).

The measurement of how well the electric power distribution system can provide a secure and adequate supply of power to satisfy the customer's requirements is called "reliability". Regarding electric power distribution systems, the electric utilities companies are responsible for the most reliable service as possible, reflecting the most advanced state of technology with reasonable cost to the end product that is the electric power[3]. Most utilities record outage information such as the number of outages, elapse time, and the number of customers interrupted. These data and statistics may be reported for each circuit or operating division, for comparison purposes, using the standard performance indices.

The performance indices provide historical *datum* which can be used to determine increasing or decreasing trends and to measure whether system improvement plans have yielded expected results.

The quality model we consider in this subsection uses the following indices, based on the sustained outage data: the SAIDI and SAIFI indexes, explained as follows:

1. SAIDI (System Average Interruption Duration Index): defined by the rate of average interruption duration per customer served per year. This index is commonly referred to as minutes of interruption per customer.

$$\text{SAIDI} = \frac{\text{Sum of Customer Interruption Durations}}{\text{Total Number of Customers Served}} \quad (29)$$

2. SAIFI (System Average Interruption Frequency Index): that defined by the rate of average number of times that a customer's service is interrupted during a reporting period per customer served in a given period (usually one year). A customer interruption is defined as one sustained interruption to one customer:

$$\text{SAIFI} = \frac{\text{Total Number of Customer Interruptions}}{\text{Total Number of Customers Served}} \quad (30)$$

It is easy to see that what is desired is a circuit with minimal SAIDI and SAIFI with the smaller cost in protective installed devices. The resulting circuit with these characteristics will the optimized circuit.

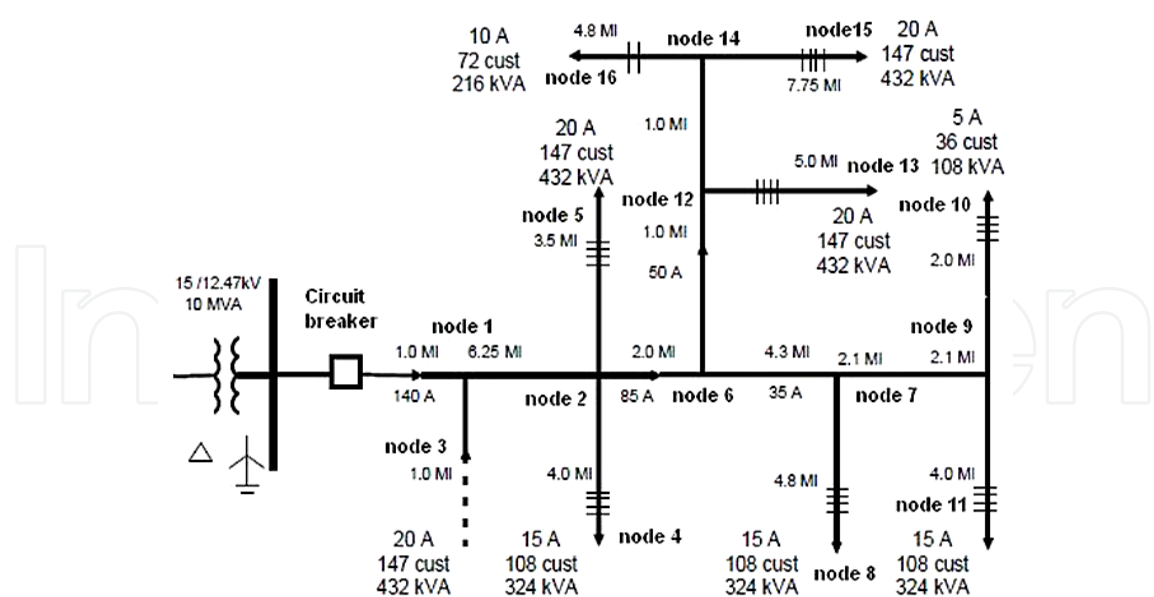[3] instead of guarantying continuous service to their customers…

Fig. 10. Circuit with Circuit Breaker in the Electric Power Substation without Reclosing Capability, based on Bishop (1997).

5.1.1 The model

The chosen model was based in the work developed by Bishop (1997) whose circuit has multiple laterals with customer's numbers and load KVA values seen on the figure 10. To perform the analysis one needs some statistics, like: number of customers; placement of protective devices on the electric power utility; good possibilities to implement protective devices; distribution circuit response to the quality indices; and traditional values of repair and recover in accordance with Bishop's indices.

The initial circuit used to the analysis is presented by figure 10, where it was considered the values of Bishop (1997) to the indices in circuits of electric power distribution with similar features in North American solutions. The used general statistical parameters are presented in table 7. As a base case analysis, the system was modelled with no reclosing of substation device. This is intended only to yield values for relative comparison with other circuits, with protective devices like recloses and fuses placed on the circuit, achieving the comparison landscape with the SAIDI and SAIFI indices.

Faults per circuit mile per year	0.22
Percent of permanent faults	20%
Percent of temporary faults	80%
Manual restoration time	2.0 hours
Repair time for 30 lines	3.0 hours
Repair time for 10 lines	2.5 hours

Table 7. General statistical parameters used in the model.

The Search for Parameters and Solutions:
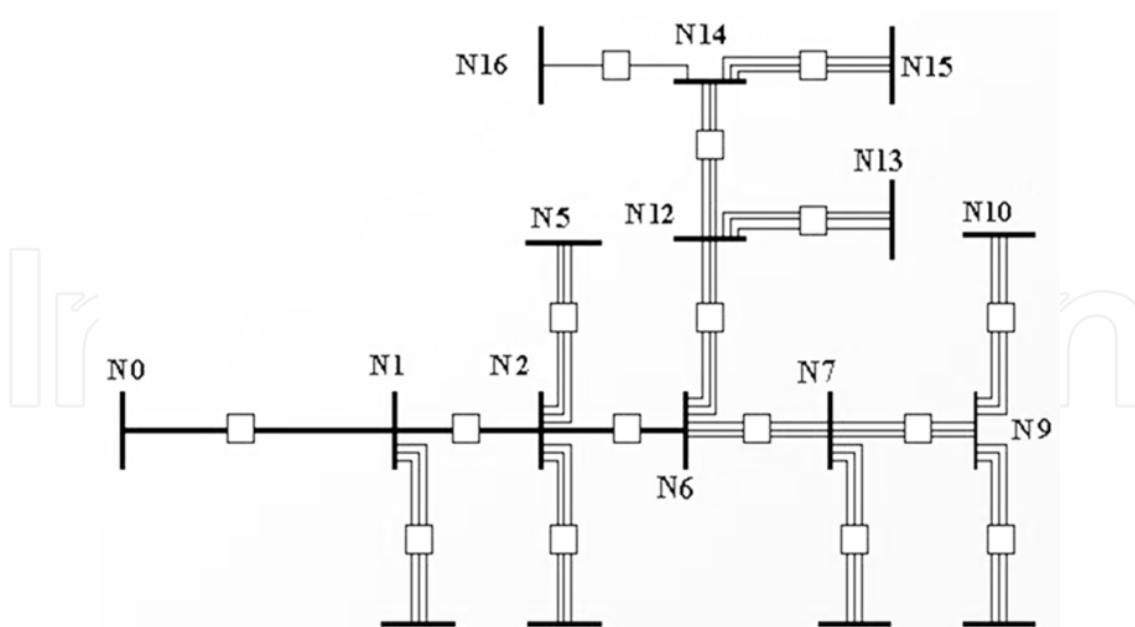Applying Genetic Algorithms on Astronomy and Engineering
183

Fig. 11. Representation of the circuit of figure 10 with the nodes with all the possible locations for protective devices (adapted from Burian et al. 2010).

5.1.2 The methodology: Converting to a GA application

The first step is to provide formalism in such a way that the protective devices net could be represented by a set of genes in a chromosome Λ, and that the Bishop (1997) model could be expressed as a P-problem whose parameters are given by Λ.

The solution chosen was to code the circuit as a series of nodes, designed by Ni, with i being an integer number, and to build a list of links between the nodes (see figure 11). The special node N0 is the main protective switch in the substation (which is present in all solutions). Each link between nodes can have a protective device, and its location is designed as $P_{i,j}$, with i and j being the two nodes that define the link. Special data structure is provided to the nodes to storage information about the number of phases, number of consumers, distance to neighbours nodes, etc.

The adopted solution considers S as a ordered list of tokens, and the position in the ordered list corresponds to a location as $P_{i,j}$. Then, for the circuit of figure 11, one has

$$\Lambda = \left\{ \begin{array}{l} P_{0,1}, P_{1,2}, P_{1,3}, P_{2,4}, P_{2,5}, P_{2,6}, P_{6,7}, P_{6,12}, P_{7,8}, \\ P_{7,9}, P_{9,10}, P_{9,11}, P_{12,13}, P_{12,14}, P_{14,15}, P_{14,16} \end{array} \right\}. \tag{31}$$

So, Λ is a finite set of tokens, and its number of elements is much smaller than the number of nodes squared[4], that assumes the role of parameter set in the P-problem. These tokens can represent a protective device to be placed in its respective circuit position. The possible devices are: main substation switch, only possible in location $P_{0,1}$ (S); fuse (F), automatic reclose switch (R) and nothing (no device).

---

[4] Of course! When representing an electric circuit one does not link one node to all the other nodes…

The kind of device defines the algorithm to be used to obtain the overall cost of protective devices, and the SAIDI and SAIFI indexes according to Bishop (1997). So, each set Λ_i represents a different circuit, and applying the Bishop's algorithms one obtains a result set

$$\Gamma_i\left(\Lambda_i\right) = \{SAIDI_i, SAIFI_i, c_{Si}, c_{Ri}, c_{fi}\} . \tag{32}$$

where c_S, c_R and c_f are the costs of the main switch reclose switch and fuses, which are expressed in monetary "units", being one unit the cost of the a monophasic fuse.

As the set Γ_i itself cannot express the degree of adaptation the individual Λ_i to the problem we want to solve, we must provide an expression to summarize Γ_i in a more convenient, single valued variable, like the *gof* value, described in subsection 3.1. The definition of this *gof* should have a monotonic behaviour as the costs and the SAIDI and SAIFI index increase. We adopted the simple expression

$$gof = \kappa_a\left(SAIDI + SAIFI\right) + \kappa_b\left(c_S + c_R + c_f\right) . \tag{33}$$

where κ_a and κ_b are constant scale converters. Then, one can say that optimized circuit will be that one that offers the smaller *gof*. With this, our inverted NP-problem can be solved by looking for the individual Λ_i that presents the smaller *gof*. As all the parameters are limited range integer numbers (tokens), some special care must be taken in the GA routines that deal with new individuals and mutation. So, these routines where rebuild taking into account the discrete character of the chromosomes. The overall behaviour of the GA optimization code follows the algorithm proposed in figure 2.

5.1.3 Results and conclusion

The resulting optimized circuit is shown in figure 12, and its corresponding indexes are presented in table 8. The GA code performed the ranging of large number of solutions and configurations, within the universe of about 50 generations of configurations. This demonstrates the GA potential in this kind of analysis and application to discrete allocation equipment's. GA optimization techniques has been showed to be an effective technique to optimize the allocation of protective devices inside the electrical distribution systems.

Index	value
SAIDI	2.7694
SAIFI	1.04385
Cost S	60 units
Number S	1
Cost R	280 units
Number R	3
Cost F	25 units
Number F	9
Total Cost	365 units

Table 8. Indexes values for optimized circuit.
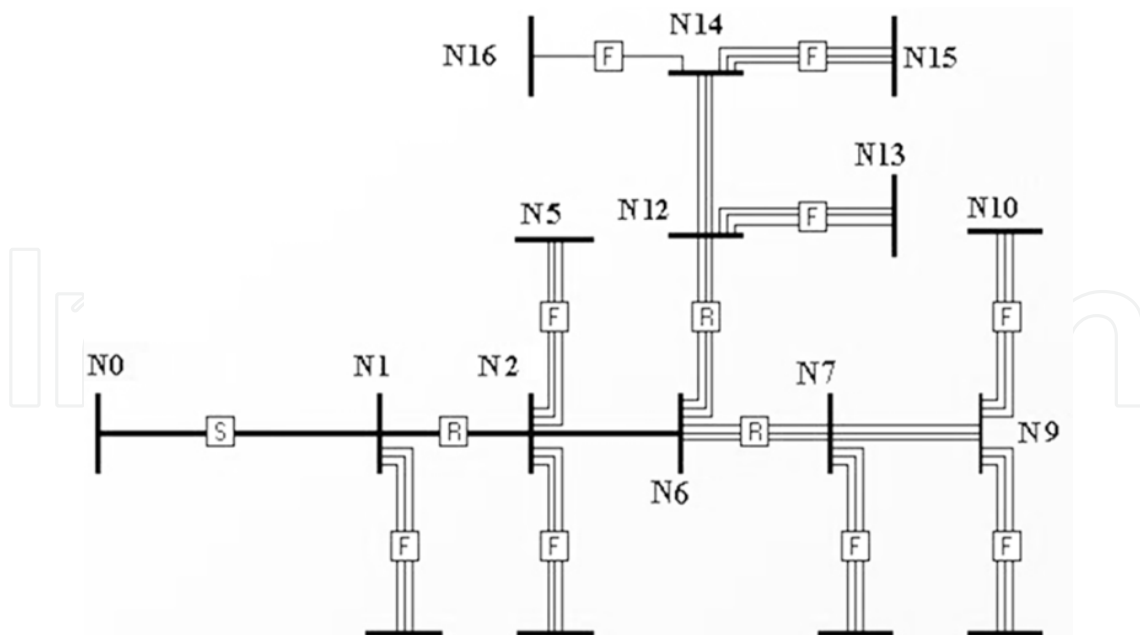
Fig. 12. Optimized circuit obtained with the GA method (adapted from Burian et al. 2010).

6. Acknowledgments

7. References

André P., Ward-Thompson D., Barsony M., (1993), ApJ, 406, 122

Bentley, P.J., & Corne D.W. (2002) *Creative Evolutionary Systems*. Morgan-Kaufmann, San Francisco.

Bishop, M.T. (March 1997) *Establishing Realistic Reliability Goals*. The Tech Advantage 97 Conference & Electric Expo.

Burian, R.; Hetem, A., Caetano, C. A. C. *Automatic Allocation of Electric Power Distribution Protective Devices* (2010) Opatija. 33rd International Convention on Information and Communication Technology, Electronics and Microelectronics. Opatija / Abbazia : IEEE, 2010. v. 1. p. 22-22.

Caetano, C.A.C., & Hetem, A. (2011) *Artificial Intelligence Parametrization of Harrington Pumps*, to be submitted to International Journal of Heat and Fluid (in preparation).

Cook, Stephen (1971) *The complexity of theorem proving procedures*. Proceedings of the Third Annual ACM Symposium on Theory of Computing. pp. 151–158.Dembczyński, J., Ertmer, W., Johann, U., Penselin, S., & Stinner, P. 1979, Z. Phys. A, 291, 207

Dixon, S. L.,Hall,C. A. (2010) *Fluid mechanics and thermodynamics of turbomachinery* 6th ed. ISBN 978-1-85617-793-1.

Dominik C., Dullemond C. P., Waters L. B. F. M., Walch S. (2003) A&A, 398, 607

Dullemond C. P., Dominik C., Natta A. (2001) ApJ, 560, 957

Garey, M. R., & Johnson, D. S. (1979) *Computers and Intractability: A Guide to the Theory of NP-completeness*. W. H. Freeman.

Goldberg D. E. (1989) *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman, Boston, MA

Gray, R.O., & Corbally, C.J. (1994) AJ, 107, 742.

Gregorio-Hetem J., Lépine J. R. D., Quast G. R., Torres C. A. O., de la Reza R. (1992) AJ, 103, 549

Grevesse, N. & Sauval, A.J. (1998) Space Science Reviews 85, 161

Griffinand, M.D., & French, J.R. (1991) *Space Vehicle Design*, AIAA.

Harrington, S. (2003) *Pistonless Dual Chamber Rocket Fuel Pump*, 39th AIAA/ASME/SAE/ASEE Hoint Propulsion Conference and Exhibit. AIAA 2003-4479.

Hetem, A., & Gregorio-Hetem, J. (2007) The use of genetic algorithms to model protoplanetary discs, MNRAS 382, 1707–1718 (2007) doi:10.1111/j.1365-2966.2007.12442.x

Hetem Jr, A. ; Gregorio-Hetem, J. (2009) *The use of Genetic Algorithms and Spectral Synthesis in the Calculantion of Abundances and Metallicities of T Tauri stars*. In: Young stars, Brown Dwarfs and Protoplanetary Disks Special Session 7 - IAU XXVII General Assembly, 2009, Rio de Janeiro - RJ. IAU XXVII General Assembly Abstract Book. Paris - France : International Astronomical Union, 2009. v. 1. p. 481-481.

Huzel, D.K. & Huang, D.H. (1967) *Design of Liquid Propellant Rocket Engines*, Rocketdyne Division, North American Aviation, Inc.

Koza J. R. (1994) *Genetic Programming II: Automatic Discovery of Reusable Programs*. MIT Press.

Kurucz, R. L. (1993) CD-ROM 13, *Atlas9 Stellar Atmosphere Programs and 2 km/s Grid* (Cambridge: Smithsonian Astrophys. Obs.)

Lada C. J., Wilking B. A. (1984) ApJ, 287, 610

Logan, E., Jr., & Roy, R, (eds) (2003) H*andbook of Turbomachinery (Second Edition Revised and Expanded)*, Marcel Dekker, Inc.

Luc, P. & Gerstenkorn, S. (1972) AA, 18, 209

Neufeld, M. J. (1995) *The Rocket and the Reich*. The Smithsonian Institution. pp. 80–1, 156, 172. ISBN 0-674-77650-X.

Papadimitriou, C. H. (1995) *Computational Complexity*. Addison-Wesley, Reading Massachusetts.

Press W. H., Teukolsky S. A., Vetterling W. T., Flannery B. P. (1995) *Numerical Recipes in C*, 2nd edn. Cambridge Univ. Press, New York

Rojas, G., Gregorio-Hetem, J., Hetem, A. (2008) MNRAS, 387, Issue 3, pp. 1335-1343.

Sutton, G.P., & Biblarz, O. (2001) *Rocket Propulsion Elements* 7th editon, JOHN WILEY & SONS, INC.

Sutton, G.P. (1986) *Rocket Propulsion Elements an Introduction to Engineering of Rockets*, John Wiley & Sons.

Torres C. A. O. (1998) *Publicação Especial do Observatório Nacional*, No. 10/99. Observatório Nacional, Rio de Janeiro

Torres C. A. O., Quast G. R., de la Reza R., Gregorio-Hetem J., Lépine J. R. D. (1995) AJ, 109, 2146

van den Ancker M. E., Meeus G., Cami J., Waters L. B. F. M.,Waelkens C. (2001) A&A, 369, 217

Wilking B. A., Lada C. J., Young E. T. (1989) ApJ, 340, 823

Young, W. C. (1989) *Roark's formulas for stress and strain*, McGraw-Hill
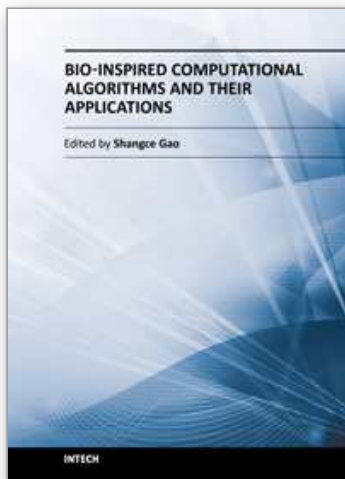
Bio-Inspired Computational Algorithms and Their Applications

Edited by Dr. Shangce Gao

Bio-inspired computational algorithms are always hot research topics in artificial intelligence communities. Biology is a bewildering source of inspiration for the design of intelligent artifacts that are capable of efficient and autonomous operation in unknown and changing environments. It is difficult to resist the fascination of creating artifacts that display elements of lifelike intelligence, thus needing techniques for control, optimization, prediction, security, design, and so on. Bio-Inspired Computational Algorithms and Their Applications is a compendium that addresses this need. It integrates contrasting techniques of genetic algorithms, artificial immune systems, particle swarm optimization, and hybrid models to solve many real-world problems. The works presented in this book give insights into the creation of innovative improvements over algorithm performance, potential applications on various practical tasks, and combination of different techniques. The book provides a reference to researchers, practitioners, and students in both artificial intelligence and engineering communities, forming a foundation for the development of the field.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Annibal Hetem Jr. (2012). The Search for Parameters and Solutions: Applying Genetic Algorithms on Astronomy and Engineering, Bio-Inspired Computational Algorithms and Their Applications, Dr. Shangce Gao (Ed.), ISBN: 978-953-51-0214-4, InTech, Available from: http://www.intechopen.com/books/bio-inspired-computational-algorithms-and-their-applications/the-search-for-parameters-and-solutions-applying-genetic-algorithms-on-astronomy-and-engineering

INTECH

open science | open minds