AIAA JOURNAL
Vol. 34, No. 2, February 1996

Chemical Laser Modeling with Genetic Algorithms

David L. Carroll*
University of Illinois at Urbana–Champaign, Urbana, Illinois 61801

A genetic algorithm technique was implemented to determine a set of unknown parameters that best matched the Blaze II chemical laser model predictions with experimental data. This is the first known application of the genetic algorithm technique for modeling lasers, chemically reacting flows, and chemical lasers. Overall, the genetic algorithm technique worked exceptionally well for this chemical laser modeling problem in a cost effective and time efficient manner. Blaze II was baselined to existing chemical oxygen-iodine laser data taken with the research assessment and device improvement chemical laser device with very good agreement. Mixing calculations for the research assessment and device improvement chemical laser nozzle indicate that higher iodine flow rates are necessary to maintain a significant fraction of the nominal performance as the total pressure is increased by the addition of helium; this agrees with research assessment and device improvement chemical laser experimental data. It may be possible to implement this genetic algorithm technique to optimize the performance of any chemical laser as a function of any of the flow rates, mirror location, mirror size, nozzle configuration, injector sizes, and other factors. This modeling procedure can be used as a method to guide experiments to improve chemical laser performance.

Nomenclature

H	= jet penetration height
L_r	= gas trip (injector) jet spacing
N_{I^*}, N_I	= number densities of I* and I
P	= pressure, torr
T	= temperature, K
U	= utilization defined as $\dot{x}_{O_2}/\dot{x}_{(Cl_2)_0}$
x_i	= mole fraction of species i
$\dot{x}_i$	= molar flow rate of species i, moles/s
Y	= yield defined as $[O_2(^1\Delta)]/[\text{total } O_2]$
β	= titration ratio defined as $\dot{x}_{I_2}/\dot{x}_{O_2}$ and $\dot{x}_{I_2}/U\dot{x}_{(Cl_2)_0}$
ψ	= diluent ratio defined as $\dot{x}_{(He)_{pri}}/\dot{x}_{(Cl_2)_0}$

I. Introduction

THE typical chemical oxygen-iodine laser (COIL) utilizes an energy transfer from the singlet delta excited state of oxygen $O_2(^1\Delta)$ directly and indirectly to I_2 to dissociate the iodine molecule. This process is followed by an energy transfer from other $O_2(^1\Delta)$ molecules to the liberated iodine atoms, thus providing the energy for the atomic iodine laser transition of interest. A number of papers have investigated issues associated with the operation of COIL systems.[1] These include the required $O_2(^1\Delta)$ generators,[2–8] the ratio of I_2 to O_2 flow rates (titration) for optimal performance,[1] the energy transfer dynamics,[9] and the power extraction efficiency,[1] among others. Crowell and Plummer[10] addressed the issue of how the extracted power is affected by significant increases in the total pressure of the flow (up to approximately 120 torr), but their power results were not in particularly good agreement with data because they used a premixed model for the laser cavity calculations. No investigations have addressed the issues of the extent of mixing, I_2 dissociation, small signal gain, power, and system performance when the total pressure of the flow is increased to levels greater than 120 torr. If the total pressure of a COIL device can be increased significantly while maintaining the device's nominal performance, then it should be possible to reduce the size of pressure recovery systems.

The effects of increased total pressure on mixing, kinetics, gain, and power must be analyzed with a reliable computer model. In Ref. 11, the mixing Blaze II computer model[12] was modified for use with a slit nozzle laser device and baselined to recent COIL data for a range of total pressures of 70–130 torr. There were five unknown parameters used in this modeling.[11] To efficiently search this large multimodal parameter space, a robust genetic algorithm (GA) technique[13,14] was implemented to find the set of unknown parameters that best matched modeling predictions with experimental data; the details of this GA technique and the implementation for a chemical laser model are the subject of this paper.

Received Jan. 21, 1995; revision received May 9, 1995; accepted for publication May 9, 1995. Copyright © 1995 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.

*Postdoctoral Research Associate, Aeronautical and Astronautical Engineering Department. Member AIAA.

II. Description of the Blaze II Model and the Unknown Parameters

The Blaze II code[12] was originally written to be as generic a chemical laser model as possible. Blaze II can treat arbitrary combinations of chemical species characterized by an adequate number of reactions (as many as 500 reactions and 40 species). Blaze II, which contains one-dimensional fluid dynamic equations whose mixing terms were derived from the two-dimensional equations that describe the mixing flowfield in a chemical laser cavity, can be used for axisymmetric and two-dimensional flows. Lasing may occur on a single atomic transition or on as many as 10 vibrational bands of a rotational equilibrium flow. Computer solution time is sufficiently short that extensive parametric studies can be performed in reasonably short times; a typical hydrogen-fluoride (HF) chemical laser run takes approximately 70 CPU seconds on an IBM RS/6000 computer, and a typical COIL run takes approximately 30 CPU seconds on the same machine. In Ref. 11 the Blaze II code was used extensively to predict the performance of a COIL (single atomic transition) device.

All of the calculations made in this paper used the 33 reaction, 12 species set[15] listed in Table 1. All of the reactions tabulated are one-way reactions. The gain g of a COIL laser is given by[1,10,15]

$$g = \frac{7}{12}\left(\frac{A\lambda^2}{8\pi}\right)\phi(\nu)\left(N_{I^*} - \frac{1}{2}N_I\right) \tag{1}$$

where $\lambda = 1.315274 \times 10^{-4}$ cm, $A = 5.0\ s^{-1}$, and the Voigt lineshape function $\phi(\nu)$ is given by

$$\phi(\nu) = \frac{2}{\Delta\nu_D}\left(\frac{\ell n 2}{\pi}\right)^{\frac{1}{2}}[1 - \text{erf}(y)]\exp(y^2) \tag{2}$$

where

$$\Delta\nu_D = \frac{2}{\lambda}\sqrt{\frac{2RT\ell n 2}{W}} = 1.4492 \times 10^7\sqrt{T} \tag{3}$$

$$y = \frac{\Delta\nu_L}{\Delta\nu_D}\sqrt{\ell n 2} \tag{4}$$

$$\Delta\nu_L = \frac{T_{ref}}{T}P\sum_i \alpha_i x_i \tag{5}$$

Table 1 Full oxygen-iodine reaction set; 33 reactions, 12 species: I, I*, I_2, I_2^*, He, H_2O, $O_2(^1\Delta)$, $O_2(^3\Sigma)$, $O_2(^1\Sigma)$, Cl_2, ICl

k								Rates, cm^3/molecule-s
1	$O_2(^1\Delta)$	+	$O_2(^1\Delta)$	→	$O_2(^1\Sigma)$	+	$O_2(^3\Sigma)$	2.5e-17
2	$O_2(^1\Delta)$	+	$O_2(^1\Delta)$	→	$O_2(^3\Sigma)$	+	$O_2(^3\Sigma)$	1.8e-17
3	$O_2(^1\Sigma)$	+	$O_2(^3\Sigma)$	→	$O_2(^1\Delta)$	+	$O_2(^3\Sigma)$	3.9e-17
4	$O_2(^1\Sigma)$	+	H_2O	→	$O_2(^1\Delta)$	+	H_2O	6.0e-12
5	$O_2(^1\Sigma)$	+	Cl_2	→	$O_2(^1\Delta)$	+	Cl_2	2.0e-15
6	$O_2(^1\Sigma)$	+	He	→	$O_2(^1\Delta)$	+	He	1.0e-17
7	$O_2(^1\Delta)$	+	$O_2(^3\Sigma)$	→	$O_2(^3\Sigma)$	+	$O_2(^3\Sigma)$	1.6e-18
8	$O_2(^1\Delta)$	+	H_2O	→	$O_2(^3\Sigma)$	+	H_2O	4.0e-18
9	$O_2(^1\Delta)$	+	Cl_2	→	$O_2(^3\Sigma)$	+	Cl_2	6.0e-18
10	$O_2(^1\Delta)$	+	He	→	$O_2(^3\Sigma)$	+	He	8.0e-21
11	I_2	+	$O_2(^1\Sigma)$	→	2I	+	$O_2(^3\Sigma)$	4.0e-12
12	I_2	+	$O_2(^1\Sigma)$	→	I_2	+	$O_2(^3\Sigma)$	1.6e-11
13	I_2	+	$O_2(^1\Delta)$	→	I_2^*	+	$O_2(^3\Sigma)$	7.0e-15
14	I_2	+	I*	→	I	+	I_2^*	3.5e-11
15	I_2^*	+	$O_2(^1\Delta)$	→	2I	+	$O_2(^3\Sigma)$	3.0e-10
16	I_2^*	+	$O_2(^3\Sigma)$	→	I_2	+	$O_2(^3\Sigma)$	5.0e-11
17	I_2^*	+	H_2O	→	I_2	+	H_2O	3.0e-10
18	I_2^*	+	He	→	I_2	+	He	4.0e-12
19	I	+	$O_2(^1\Delta)$	→	I*	+	$O_2(^3\Sigma)$	7.8e-11
20	I*	+	$O_2(^3\Sigma)$	→	I	+	$O_2(^1\Delta)$	1.0277e-10*exp(−401.4/T)
21	I	+	$O_2(^1\Delta)$	→	I	+	$O_2(^3\Sigma)$	1.0e-15
22	I*	+	$O_2(^3\Sigma)$	→	I	+	$O_2(^3\Sigma)$	3.5e-16
23	I*	+	$O_2(^1\Delta)$	→	I	+	$O_2(^1\Sigma)$	1.0e-13
24	I*	+	$O_2(^1\Delta)$	→	I	+	$O_2(^1\Delta)$	1.1e-13
25	I	+	I*	→	I	+	I	1.6e-14
26	I*	+	H_2O	→	I	+	H_2O	2.1e-12
27	I*	+	He	→	I	+	He	5.0e-18
28	I*	+	Cl_2	→	Cl	+	ICl	5.5e-15
29	I*	+	Cl_2	→	I	+	Cl_2	8.0e-15
30	I*	+	ICl	→	I_2	+	Cl	1.5e-11
31	I_2	+	Cl	→	I	+	ICl	2.0e-10
32	Cl	+	ICl	→	I	+	Cl_2	8.0e-12
33	I_2	+	2I	→	I_2	+	I_2	3.6e-30

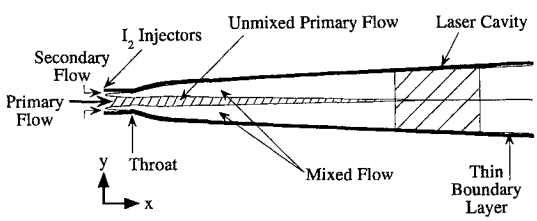

Fig. 1 Side view schematic of the RADICL nozzle showing the laser cavity location and the mixed and unmixed flow regions.

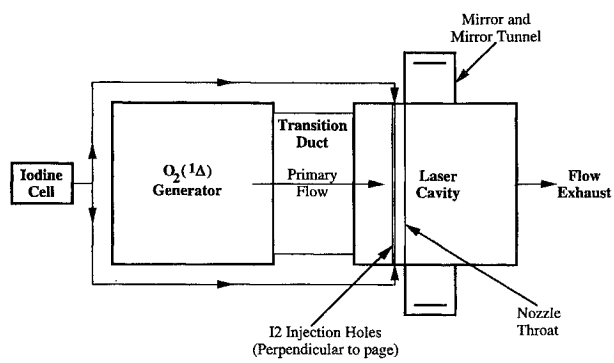

Fig. 2 Top view schematic of the RADICL experimental layout.

where R is the universal gas constant, 8.3143×10^7 ergs/mole-K; W is the molecular weight of iodine, 126.90447 gm/mole; T_{ref} is a reference temperature taken to be 295 K; and the pressure broadening coefficients α_i in MHz/torr (referenced to 295 K) are 4.4, 10.5, 7.3, 6.2, 20.6, 3.6, 3.6, and 4.7 for I, I_2, O_2, Cl_2, H_2O, He, N_2, and Ar, respectively.[10]

Since the high-pressure COIL experiments have been performed with the research assessment and device improvement chemical laser (RADICL) nozzle at Phillips Laboratory, the modeling investigation focused on that nozzle. A side view schematic of the RADICL nozzle and laser cavity section are shown in Fig. 1. A top view schematic of the RADICL device is shown in Fig. 2. Detailed discussions of the experimental configuration are given in Refs. 10 and 11. The data modeled have a nominal chlorine flow rate into the oxygen generator of $\dot{x}_{(Cl_2)_0} = 0.50$ moles/s. The primary flow rate of He is given by the diluent ratio ψ [$=\dot{x}\text{He}/\dot{x}_{(Cl_2)_0}$]; the primary flow rate of O_2 was equal to $U\dot{x}_{(Cl_2)_0}$ where the utilization U was given by experimental measurements, approximately 0.93 for $\psi = 3$ and 0.86 for $\psi = 10$; the secondary flow rate of iodine is given by the titration ratio β [$=\dot{x}_{I_2}/\dot{x}_{O_2} = \dot{x}_{I_2}/U\dot{x}_{(Cl_2)_0}$]; and the secondary flow rate of He was equal to $0.65 + 0.10(\psi - 3)$ moles/s (Ref. 11).

Blaze II requires the following inputs for each stream: the total mass flow rate, the mass fraction of each species, the pressure, the temperature, the Mach number, the initial height of the stream, and the reference diffusion coefficient. With the exception of the oxygen yield [which gives the relative mass fractions of $O_2(^1\Delta)$ and $O_2(^3\Sigma)$] and water vapor in the primary stream at the I_2 injectors, all of these initial conditions for Blaze II are accurately measured experimentally (mass flow rates) or can be approximately determined from experimentally measured flow conditions.[11] There are experimental values for the oxygen yield; unfortunately, this yield diagnostic is not very accurate because of calibration difficulties. Hence, there is a great deal of uncertainty in the oxygen yield in the plenum of the RADICL nozzle; experimental values range from 0.22 to 0.55

(Refs. 16–18). The functional dependence of the yield on the diluent ratio ψ is illustrated in Fig. 6 of Ref. 11. The amount of water vapor is believed to be in the range of 0.01–0.10 moles/s and approximately constant as a function of the primary flow rates.[16] Therefore, a study of the effects of changing the yield and the amount of water vapor was undertaken.[11] The objective of that investigation was to determine a combination of the yield and water vapor parameters that best predicted the experimental outcoupled power over the entire data set.

Unfortunately, it was found in Ref. 11 that by varying only the yield and water vapor, it was impossible to produce good agreement with data for different mass flow rates. Therefore, to obtain better agreement with the data as the mass flow rates were increased, two mixing parameters were added to the list of unknown parameters. First, for generality it was assumed that the I_2 injector jets expand by an average factor J_{exp} when they enter the lower pressure primary flow. The jets will definitely expand, but the amount of expansion is uncertain. Reference 19 indicates that an underexpanded jet with a pressure ratio $[(P_0)_{sec}/P_{pri}]$ of approximately 6 [=(300 torr)/(50 torr), the ratio for the RADICL experimental conditions] has an expansion of approximately unity, i.e., very little expansion. If we consider simple quasi-one-dimensional fluid dynamic expansion for a $\gamma = 1.67$ fluid ($\gamma \approx 1.65$ for the He/I_2 secondary jets) from Mach 1 to a pressure ratio of $P_{pri}/(P_0)_{sec} \approx 1/6$, the area ratio A/A^* is $\approx$1.32. Since the jet expands three dimensionally, the effective diameter of the expanded jet goes as the square root of the area ratio, $D_{exp}/D \approx 1.15$. The jet, however, would expand to this value over the full penetration height; therefore, the average expansion is only around $1.075 \approx 1.1$. Thus, the a priori assumption was that $J_{exp} = 1.1$. Since J_{exp} is actually an unknown, however, it was allowed to have a range in the parameter space; since the jets expand, the minimum value is 1.0 and a value of 2.5 was arbitrarily designated the maximum.

Second, to simulate a three-dimensional mixing flow with a two-dimensional mixing model, the reference diffusion coefficients for the primary and secondary streams incorporate a diffusion coefficient multiplier (DCM). This DCM is based on a surface stretching scheme proposed by Driscoll,[20,21] who suggests that an appropriate DCM is

$$DCM \approx \psi_r^2 = 1 + (H/L_r)^2 \tag{6}$$

where ψ_r is a constant reference value of the surface extension ratio. For these modeling calculations, the jet penetration height is found from the momentum flux ratio of the secondary to the primary streams and the study of Cohen et al.[22] of jet penetration (the value of H is taken to be H_{top} defined in Ref. 22). There are two rows of I_2 injectors in the plenum region of the RADICL nozzle, Figs. 1 and 2. There are 115 large and 230 small injector holes on each side (top and bottom) of the nozzle plenum. The large and small I_2 injectors have geometric diameters of 0.032 and 0.016 in., respectively. The large holes have a spacing of 0.085 in. between their centerlines and the small holes have a spacing of 0.0425 in. between their centerlines. The centerline of the row of large holes is 0.12 in. upstream of the centerline of the row of small holes. The centerline of the row of large injectors is considered to be $x = 0$ in the Blaze II modeling calculations. Because of the geometry of the problem (large injectors are twice the diamater of the small injectors, but there are twice as many small injectors), the DCM for each set of injectors is the same. The surface stretching mixing enhancement scheme is represented by an increase in the contact area between the two streams. Because the primary flow, which passes between the large injector jets, should also pass between and over the small injector jets, the a priori assumption was that the enhanced mixing effects of the set of large injectors should not be significantly improved by the second set of injectors, i.e., only the large set of injectors was a priori considered to enhance the mixing. To simulate the effects of further enhanced mixing by the secondary injectors, the DCM was increased by an arbitrary exponent

$$DCM = (DCM)^{DCM_{exp}} \tag{7}$$

where the DCM_{exp} was allowed to range between 1.0 and 2.5. Since the DCM of the large and small injectors are the same, it is plausible that the effective DCM could be $(DCM_{large})(DCM_{small}) = (DCM_{large})^2$; the potential for a squared relationship motivated the exponential form of the DCM used in Eq. (7). Multiplying the nominal laminar diffusion coefficients (calculated from diffusional transport theory[23]) by the DCM gives the necessary reference diffusion coefficients for the primary and secondary streams.

It was found in Ref. 11 that by varying the four unknown parameters, yield, water vapor flow rate, J_{exp} and DCM_{exp}, it was still impossible to produce good agreement with data for different mass flow rates. It is possible to argue that the Blaze II quasi-two-dimensional mixing scheme is inadequate to model the performance of a laser with a three-dimensional flowfield. However, since two of the parameters that were investigated dealt with mixing, it seems unlikely that the quasi-two-dimensional mixing model of Blaze II could be the only factor preventing the model from giving reasonable agreement with data. In fact, the Blaze II model has been extensivley used in the past to model the performance of complicated HF mixing lasers as a function of different flow rates with quite reasonable agreement[21,24,25]; this suggests that there may be some other mechanism preventing agreement with the RADICL data. The problem was related to increased total pressure by the addition of He. This raised the question of whether a chemical rate constant for a He reaction should be adjusted. The only rate constant involving He which is significantly large is that for $I_2^* + He \rightarrow I_2 + He$, rate k_{18} in Table 1. When the agreed upon COIL kinetics package in Ref. 26 was consulted, it was found that this reaction's rate constant had a low degree of confidence. Therefore, any adjustments to this rate constant within an order of magnitude should not be unreasonable. Thus, a multiplier of the rate constant k_{18} was added as a fifth parameter and allowed to range between 1 and 16. This parameter was found to be the most important parameter for baselining the model to experimental data as a function of the He flow rate.[11]

The question then becomes, what combination of these five parameters would provide good overall agreement? Thus began an extensive search for a parameter set that minimized the differences between the predicted power and the data. The parameters were discretized into increments in the aforementioned ranges, Table 2. When the number of possibilities for each parameter are multiplied together, we are left with five parameters having some 2×10^6 possible permutations. To make the situation more complicated, preliminary calculations (using different sets of parameters, which were believed to be reasonable guesses) showed that the parameter space was highly multimodal, i.e., there are a very large number of local minima in this parameter space.

Table 2 Genetic algorithm parameter space for the Blaze II modeling

Parameter	Yield	H_2O, moles/s	k_{18} multiplier	Jet expansion factor, J_{exp}	DCM_{exp}
Range	0.34–0.65	0.01–0.16	1–16	1.0–2.5	1.0–2.5
Number of possibilities (the cardinality for floating point coding)	32	16	16	16	16
Number of binary digits (parameter length for binary coding)	5	4	4	4	4
Increment	0.01	0.01	1	0.1	0.1

III. Application of Genetic Algorithms to Chemical Laser Modeling

The question now becomes, how do we efficiently search this highly multimodal parameter space for a combination of these five parameters that provide good overall agreement with data? In other words, what method should we use to efficiently search this highly multimodal parameter space for a global minimum? To answer this question we will briefly follow the reasoning and arguments in Chap. 1 of Ref. 14.

> The current literature identifies three main types of search methods: calculus-based, enumerative, and random.... First, [calculus-based] methods are local in scope; the optima they seek are the best in a neighborhood of the current point.... Second, calculus-based methods depend upon the existence of derivatives (well-defined slope values). Even if we allow numerical approximation of derivatives, this is a severe shortcoming.... The real world of search is fraught with discontinuities and vast multimodal, noisy search spaces.... It comes as no surprise that methods depending upon the restrictive requirements of continuity and derivative existence are unsuitable for all but a very limited problem domain. For this reason and because of their inherently local scope of search, we must reject calculus-based methods. They are insufficiently robust in unintended domains.
>
> [Enumerative schemes look] at objective function values at every point in the [search] space, one at a time... such schemes must ultimately be discounted in the robustness race for one simple reason: lack of efficiency.... Even the highly touted enumerative scheme dynamic programming breaks down on problems of moderate size and complexity.
>
> Random search algorithms have achieved increasing popularity as researchers have recognized the shortcomings of calculus-based and enumerative schemes.... The genetic algorithm is an example of a search procedure that uses random choice as a tool to guide a highly exploitative search through a coding of a parameter space. Using random choice as a tool in a directed search process seems strange at first, but nature contains many examples. Another currently popular search technique, simulated annealing, uses random processes to help guide its form of search for minimal energy states.... The important thing to recognize at this juncture is that randomized search does not necessarily imply directionless search.

Since the Blaze II chemical laser model is highly nonlinear and the large parameter space (with approximately 2×10^6 permutations) is highly multimodal, calculus-based and enumerative techniques were discounted as being either not robust enough or not efficient enough to handle this problem. Thus, the two most efficient algorithms that are robust enough to search for a global minimum given this highly nonlinear chemical laser model in this highly multimodal phase space (many local minimum) are the genetic algorithm[13,14] and the simulated annealing technique.[27] Advantages and disadvantages of these techniques are discussed in Refs. 27 and 28. Sufficient comparisons have not yet been made between the two techniques, and there was no clear reason to use one method over the other; thus, the genetic algorithm was chosen for its biological and evolutionary appeal.

A brief description of the GA technique will be presented, followed by a more detailed description of the technique applied to the chemical laser problem; for an excellent in-depth treatment of the subject of GAs, Ref. 14 is highly recommended. A genetic algorithm operates on the Darwinian principle of "survival of the fittest." An initial population of size n is created from a random selection of the parameters in the parameter space. Each parameter set represents the individual's chromosomes. Each of the individuals is assigned a fitness based on how well each individual's chromosomes allow it to perform in its environment. There are then three operations that occur in GAs to create the next generation: 1) selection, 2) crossover, and 3) mutation. Fit individuals are selected for mating, whereas weak individuals die off. Mated parents create a child with a chromosome set that is some mix of the parent's chromosomes. For example, parent 1 has chromosomes abcde, whereas parent 2 has chromosomes ABCDE; one possible chromosome set for the child is abcDE, where the position between the c and D chromosomes is the crossover point. Then there is a small probability that one or more of the child's chromosomes will be mutated, e.g., the child ends up with chromosomes abcDF. The process of mating and child creation is continued until an entirely new population of size n is generated with the hope that strong parents will create a fitter generation of children; in practice, the average fitness of the population tends to increase with each new generation. The fitness of each of the children is determined, and the process of selection/crossover/mutation is repeated. Succesive generations are created until very fit individuals are obtained. To assist the visualization of the GA procedure, a logic flow diagram is presented in Fig. 3.

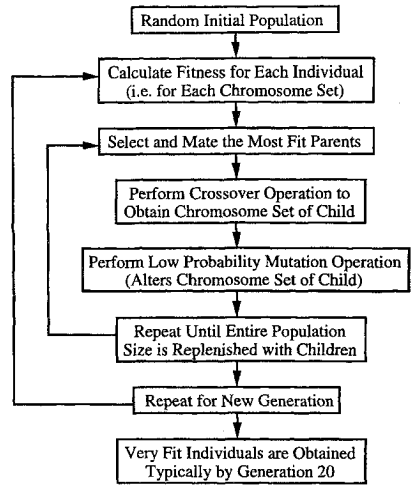

Fig. 3 Logic flow diagram for a typical genetic algorithm.

When the GA was applied to the chemical laser problem, the initial population of size n was chosen from a random selection of the parameters listed in Table 2; $n = 100$ or 200 worked well for this GA application. Random numbers were generated using Knuth's subtractive method.[29] Knuth's algorithm is regarded as one of the best random number generators. Each individual i has one set of the five parameters. The fitness of each individual is

$$\text{fitness}(i) = \frac{1}{\sum_m |(\text{power})_{\text{Blaze II}} - (\text{power})_{\text{data}}|} \tag{8}$$

where m is the number of Blaze II calculations being compared to m different data points. Note, low total power differences represent high fitnesses (strong individuals).

The five data points used initially (the last two were altered slightly in later GA trials, discussed later) were 1) $\psi = 3$, $\beta = 0.016$, 4400 W, 2) $\psi = 6$, $\beta = 0.016$, 4600 W, 3) $\psi = 6$, $\beta = 0.025$, 200 W, 4) $\psi = 10$, $\beta = 0.035$, 3000 W, and 5) $\psi = 10$, $\beta = 0.040$, 3000 W. Other sets of data points can be appropriate for this modeling problem, but arbitrarily chosen sets would not be appropriate in general. It is important that a broad range of flow conditions producing high and low power be chosen, otherwise the GA may find a good match to a localized region of the data, which may not be a good match to the overall data set. For example, it was possible to obtain a good match to data points 1, 2, 4, and 5 without data point 3, but then the overall match to data did not predict the fall off of power as ψ was increased at a constant β (power vs ψ curves are shown in Fig. 8). One reason for minimizing the number of data points to be matched is that when baselining to larger numbers of power data (six or more points), it was found that a single large error in matching one data point could average out to a reasonably small overall error when the other data were very well matched; this represents a situation where the average error may be acceptable, but the overall match to data was poor. Although not implemented in this study, it should be possible to create a more restrictive fitness function than Eq. (8) to alleviate this problem, e.g., a fitness function that would penalize a parameter set more heavily if it had a very poor match with any given data point. Since each data point tested represents an entire Blaze II power calculation, the total computer time is proportional to the number of data points tested. Preliminary trial and error calculations indicated that fewer than five data points would not cover a broad enough range of flow rates and resulting powers, that five would be adequate, and that more than five could result in the aforementioned problem and would not significantly

improve accuracy without significantly increasing run time. Thus, to minimize required computer time while retaining a broad range of conditions, the representative set of five data points ($m = 5$) covering a broad range of flow rates, pressures, and powers were chosen to test the overall fitness of each parameter set.

There were two parameter coding schemes (floating point and binary), many selection schemes, and different types of crossover and mutation operators that have been investigated in previous GA work (for an outline of previous GA work, see Ref. 14). For this application, the best choice of coding/selection/mutation schemes was unknown, so the most promising choices were investigated. For completeness, a brief outline of each of the tested options follows.

Floating point coding: the parameter is descritized into a number of possibilities, and there is one chromosome for each parameter. The value of the chromosome is stored as a floating point number. The value of the parameter is called an allele. For this application, floating point coding produced a total chromosome string length of 5. The length of each parameter string is called the parameter length, e.g., for floating point coding, each parameter is represented by a single chromosome in the chromosome string; therefore, for floating point coding, the parameter length is unity.

Binary coding: the parameters are descritized into a number of possibilities, but the chromosome length is based on the total number of possibilities in a binary format, e.g., 32 possibilities would be represented by a string of five 0s and 1s, whereas 16 possibilities would be represented by a string of four 0s and 1s. For this application, binary coding produced a total chromosome string length of 21. Again, the length of each parameter string is called the parameter length. For binary coding, each parameter is represented by a string of 0s and 1s in the total chromosome string; therefore, for binary coding, the parameter length is 5 for 32 possibilities and is 4 for 16 possibilities, Table 2. During crossover with binary coding, the crossover point may occur in the middle of one of the parameter strings; this allows the child to have a parameter string that is a mix of the parents parameter strings and, consequently, the child may have an allele (parameter value) between the two alleles of the parents. In floating point coding, the child must have a mix of the parents' alleles but cannot have alleles which are not present in the parents chromosome strings. Thus, in binary coding, more alleles (possible values of the parameters) are preserved as new generations are created.

Expected value selection: the fitness of all of the individuals in the population is summed, and then the expected probability of selection is based on the fitness of the individual divided by the total fitness of the population, i.e., $p_i = f_i / \Sigma_i f_i$. The expected number of parents with chromosome set i for the new generation is simply np_i. This procedure will fill most of the parent slots, but there will be a fractional remainder of slots that are filled using the stochastic remainder sampling without replacement method (see Ref. 14). Random pairs of mates are then chosen from this population of fit parents. Then, each pair of mates creates two children, e.g., one child could end up with chromosome set abcDE and the other with ABCde.

Tournament selection: random pairs are selected from the population and the stronger (most fit) of each pair is allowed to mate. Each pair of mates create one child, which has some mix of the two parents chromosomes according the method of crossover (discussed next). The process of selecting random pairs and mating the stronger individuals continues until a new generation of size n is repopulated.

Single-point crossover: the chromosome set of the first parent is mapped into the child, e.g., abcde. A crossover point is randomly chosen where the chromosome set of the second parent, ABCDE, overwrites the chromosome set of the first parent, e.g., one possible chromosome set for the child is abcDE, where the position between the c and D chromosomes is the randomly determined crossover point. For this study, the probability of a single-point crossover occuring, p_{cross}, was set at 0.6 (a choice suggested by De Jong; see Ref. 14). Note that there is a $1\text{-}p_{\text{cross}}$ probability that the child would retain the entire chromosome set of the first parent.

Uniform crossover: in uniform crossover, it is possible to obtain any combination of the two parents chromosomes, e.g., the child could end up with chromosome set aBcDe. For this study, the probability for a crossover occuring at each chromosome position was set at 0.6 (the same value used for single-point crossover). Note that it is possible that the child could retain the entire chromosome set of either parent, but in uniform crossover it is unlikely.

Jump mutation: there is a small probability that one or more of the child's chromosomes will be mutated, e.g., the child ends up with chromosomes abcDM, where M was not a chromosome from either parent. The jump mutation produces a chromosome that is randomly picked to be in the range of the appropriate parameter, e.g., the parameter could jump from one side of the range to the other side. The probability of a jump mutation occurring for each chromosome was set equal to $1/n$, i.e., the inverse of the population size. The choice of a $1/n$ is based upon De Jong's suggestion from his study of GAs in function optimization (see Ref. 14). The jump mutation operator was used in all of the GA work in this study.

Creep mutation: another small probability mutation is that one or more of the child's parameters will be mutated by a single increment, e.g., the child ends up with chromosomes abcDF, where F was not a chromosome from either parent, but is only one increment away from parent 2's chromosome value of E. The creep mutation produces a parameter value that is randomly picked to be larger or smaller, so long as it remains in the range of the appropriate parameter, i.e., the parameter could creep one increment up or down from one of the parents values. The probability of a creep mutation occurring for each chromosome was set equal to $1/n$, i.e., the inverse of the population size (a probability of $1/n$ was chosen for the same reason it was used for jump mutations).

Elitism: this operator is used to ensure that the chromosome set of the best parent generated to date is reproduced. After the population is generated, the GA checks to see if the best parent has been replicated; if not, then a random individual is chosen and the chromosome set of the best parent is mapped into that individual. Although this operator is not necessary, it was found to help prevent the random loss of good chromosome strings.

The simple GA described in Ref. 14 used binary coding, roulette wheel (stochastic) selection, and jump mutations. The basic GA of this study also used jump mutations, but expected value selection (an improved selection technique[14]) was used, and floating point coding was chosen for early studies for convenience. Unless otherwise noted in the text or figures, binary coding, tournament selection, uniform crossover, creep mutations, and elitism were not used for a given test.

The Blaze II model was automated for many calculations based on the five parameters, and the genetic algorithm technique was coded and implemented to search the parameter space and minimize the differences between the predicted and experimental powers. Approximately 4 days of continuous running on a RS/6000 was required to find a minimum for these five data points for a population size of 100; 8 days for a population size of 200. Since it would take too much computer time to find a global minimum for the entire data set (43 data points), the best result found for the five data points is probably not the global minimum, but it is believed to be reasonably close. This is the first known application of the genetic algorithm technique for modeling lasers, chemically reacting flows, and chemical lasers.

To better understand the GA technique and how it is applied to this chemical laser modeling problem, several different trials were run for different population sizes, coding schemes, selection schemes, and mutation types. The effects of adding elitism to the GA were tested. To judge how well the GA was matching the data, the fitness, Eq. (8), was converted to an error based on the total power of the five data points,

$$\text{error} = \sum_m |(\text{power})_{\text{Blaze II}} - (\text{power})_{\text{data}}| \Big/ \sum_m (\text{power})_{\text{data}} \quad (9)$$

Two errors were considered: the overall average error of all of the members of the population and the best individual with the smallest error. The average error is a measure of how well the population as a whole is doing, as well as how fast it is converging to the optimal solution. The best error simply indicates how well the GA has done in finding a minimum cost solution.

Before any GA trials were run, a series of trial and error calculations were performed in an effort to isolate the global minimum. The five data points used were 1) $\psi = 3$, $\beta = 0.016$, 4400 W, 2) $\psi = 6$, $\beta = 0.016$, 4600 W, 3) $\psi = 6$, $\beta = 0.025$, 200 W, 4) $\psi = 10$, $\beta = 0.035$, 3000 W, and 5) $\psi = 10$, $\beta = 0.040$, 3000 W. These five data points represent nominal flow rates that experiments attempted to obtain. In fact, the experiments did not obtain precisely these flow conditions (in particular, the two $\psi = 10$ nominal data conditions were difficult to obtain precisely); the effects of using exact flow conditions are discussed later. The listed power numbers for these five data points are averaged values over actual data with approximately these flow conditions. The lowest total power difference that could be obtained was 1396 W, which was equivalent to an error of 0.093 (9.3%). These trial and error calculations required approximately 2 weeks of effort to perform. In essence, the criteria for whether or not the GA technique was of use were twofold: 1) Did the GA arrive at a better solution than the trial and error method? 2) Did the GA take less time to arrive at this solution? As will be shown, the answer to both of these questions was an emphatic yes.

The question of what is an appropriate population size must be addressed before any GA calculations can be run. From Ref. 30, an appropriate population size is

$$n = \mathcal{O}(m\chi^k) = \mathcal{O}[(\ell/k)\chi^k] \tag{10}$$

where $m = \ell/k$; χ is the cardinality of chromosomes, i.e., the number of possibilities for each chromosome, e.g., for binary coding $\chi = 2$; k is the size of the schema[13,14] of interest, and ℓ is the length of the chromosome string. From Ref. 30, a schema can be defined as "a similarity template describing a subset of strings with similarities at certain string positions." A schema can be thought of as some combination of the chromosomes in the complete chromosome string, e.g., one possible schema in a string of five 0s and 1s would be 1*0**, where the asterisk denotes a value which is unimportant to the particular schema; the length of the schema of this example would be 3, i.e., 1*0 has three places. Another possible schema would be 11**0, which has a schema length of 5.

For the purposes of estimating an appropriate population size, it was assumed that each parameter string would represent one important schema; therefore, the schema length was assumed to be equal to the parameter length for estimation purposes. For floating point coding, each chromosome is of interest, so the schema length k is 1. The total chromosome string length is 5 (one for each of the five parameters), and the cardinality can be approximated as the average cardinality of the chromosomes $\chi \approx (1/5)(32 + 16 + 16 + 16 + 16) \approx 20$. So, for floating point coding, the population size should be on the order of $(5/1)(20^1) = 100$. For binary coding, $\ell = 21$, $\chi = 2$, and the average length of the schema of interest is $k \approx (1/5)(5 + 4 + 4 + 4 + 4) \approx 4$ (this is the average length of the number of chromosomes that make up one parameter), so the population size should be on the order of $(21/4)(2^4) = 84$. Thus, population sizes of 50, 100, and 200 were tried for this problem.

The first GA trials were made with a population size of 100 for five data points. The five data points used were the same as those used in the trial and error calculations. The average and best (minimum) errors as a function of the generation for six different GA techniques are plotted in Figs. 4 and 5. The first two GA techniques were simple expected value trials with two different numbers used to initialize the random number generator. The average error of both of the expected value trials decreased as more generations were created. The first trial (random 1) began to oscillate around an average error of approximately 0.12 starting at generation 12; this was indicative of the fact that the GA had narrowed the number of alleles (the actual values of the parameters) to a very few, such that further significant improvements would require a lucky random mutation. Regardless of the fact that the first trial had lost most of the available alleles, it still obtained a best error of 6.6% (Fig. 5), which was better than that obtained by two weeks of trial and error calculations. This was encouraging. Since elitism was not a requirement of the second trial (random 2), it can be seen that the best error increased from generation 6 to generation 7 (Fig. 5); because of the subsequent loss of alleles as the number of generations increased, the best error never fell below 10% thereafter. Thus, to prevent the loss of the best individual during later generations, elitism was invoked for all subsequent GA trials.

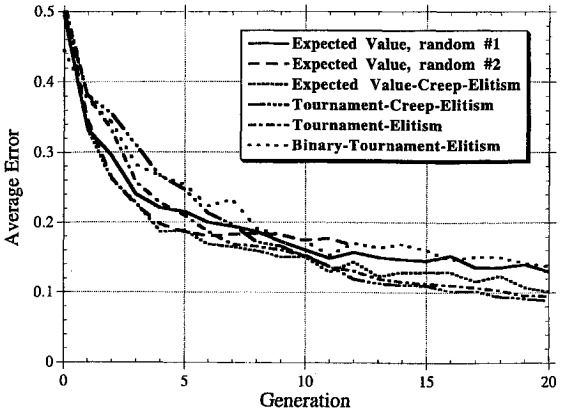

Fig. 4 Average error as a function of the generation for different genetic algorithm schemes; population size for all cases was 100.

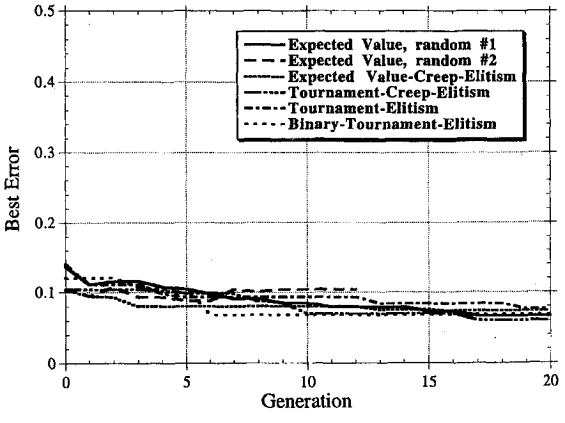

Fig. 5 Best (lowest) error as a function of the generation for different genetic algorithm schemes; population size for all cases was 100.

Discussions with Goldberg[31] indicated several methods of reducing the loss of alleles and improving the performance of the GA. The next trial run was to add the creep mutation and elitism to the GA (Figs. 4 and 5). These additions did not significantly improve the performance of the GA but did ensure that the best individual from each generation survived. Tournament selection was then substituted for the expected value selection, while the creep mutation and elitism were retained; this combination produced the best results for the selected five data points (Figs. 4 and 5). The average error initially decreased more slowly but continued to decrease and did not oscillate like the previous three trials (Fig. 4); these attributes are indicative of preserving more alleles during the first several generations, which makes for a more efficient and global search procedure. The best error with the tournament-creep-elitist scheme was 6.0%. To test the effects of the creep mutation, it was subtracted from the previous scheme; although the average error was similar to the tournament-creep-elitist scheme (Fig. 4), the best error ended at 7.5%, which was not as good as the previous scheme (Fig. 5). Binary coding for the chromosomes was then added to the tournament-elitist scheme. Of all of the schemes, this binary-tournament-elitist scheme showed the greatest preservation of alleles (Fig. 4). In addition, the binary-tournament-elitist scheme gave a best error of approximately 7.0%, which is better than the tournament-elitist scheme (Fig. 5). At this point, the tournament-creep-elitist scheme was the overall best for this problem.

The next trials tested the effects of population size. To decrease the amount of required computer time (4 days on an IBM RS/6000 machine for a population size of 100), the GA performance for a population size of 50 was determined. Based on the previous $n = 100$ trials, two tests were run: expected value and tournament-creep-elitist (Fig. 6). As was the case for $n = 100$, the simple expected value scheme only performed moderately well but did produce a best error of 8.2%, which was better than the 9.3% of the trial-and-

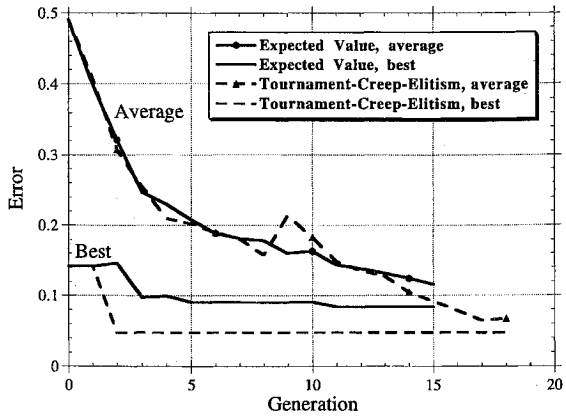

Fig. 6 Error as a function of the generation for different genetic algorithm schemes; population size for all cases was 50.

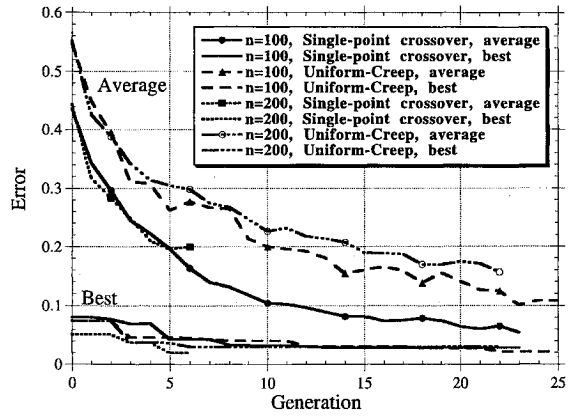

Fig. 7 Error in the GA modeling of data as a function of the generation, population size and type of crossover operator (single-point or uniform) for a binary-tournament-elitist GA.

error calculations. The tournament-creep-elitist scheme produced excellent results; the best error was 5.0% (Fig. 6). An obvious question arises here, why did the $n = 50$ case produce an error less than the $n = 100$ case? The answer is chance. Since the individuals are essentially generated randomly under the constraint of certain selection rules, there is no reason why a smaller population could not produce a more fit individual than a larger population. It is less likely, however, that the smaller population would produce a more fit individual. The fact that an $n = 50$ trial produced a more fit individual than an $n = 100$ trial demonstrates that the GA technique does not guarantee finding a global minimum to this problem. The probability of finding a global minimum, however, will increase with population size.[30] A very important feature about the $n = 50$ cases is that the average error very rapidly approaches the best error (Fig. 6); i.e., for this small population, the available alleles are very rapidly removed as the more fit individuals take over. For this important reason, $n = 50$ was judged to be too small a population size to adequately search the entire parameter space, despite the chance occurance of a smaller best error than in the $n = 100$ case.

An important point is that for the sake of keeping computer run time down to tolerable levels, the GA was stopped at generation 20, rather than allowed to continue searching for more fit individuals. This is a factor that can contribute to the fact that the $n = 100$ trials did not produce as fit an individual as an $n = 50$ trial. A detailed look at the remaining alleles for the $n = 50$ and 100 trials showed that many more alleles still remained for the $n = 100$ trials than for the $n = 50$ trials. This is an indication that further generations with $n = 100$ still have a reasonable probability of producing a more fit individual than that obtained at generation 20; this phenomenon can be seen to be true in Fig. 7 for the $n = 100$, uniform-creep case (discussed in more detail later), i.e., for this case, a smaller error was found at generation 22. Since most of the alleles were gone by generation 20 for the $n = 50$ trials, the probability of finding more fit individuals beyond generation 20 is very low.

When the best sets of parameters obtained from the previous GA trials were used for all of the available data (43 data points), the model was in reasonable agreement with data. For the high β calculations ($0.030 < \beta < 0.045$), however, the model generally predicted too little power for $0.030 < \beta < 0.035$ and too large a power for $0.040 < \beta < 0.045$ (not shown). When looking at the actual data runs for these high-β data points, it was found that there were some significant variations in the flow rates of different gases (especially the secondary He flow rate), which were not accounted for in the model. As such, the five data points were changed to 1) $\psi = 3$, $\beta = 0.016$, 4400 W, 2) $\psi = 6$, $\beta = 0.016$, 4600 W, 3) $\psi = 6$, $\beta = 0.025$, 200 W, 4) $\psi = 10.33$, $\beta = 0.0317$, 3050 W, and 5) $\psi = 9.64$, $\beta = 0.0418$, 3080 W (note that only the fourth and fifth data points were changed). Subsequent GA calculations modeled these five actual data points in an effort to improve agreement between the model and the data.

Based on the previous GA calculations, the tournament-creep-elitist and binary-tournament-elitist schemes worked best for this application. In Ref. 32 it was shown that floating-point coded GAs can be "blocked" from finding the global minimum because important alleles can be lost as separated local minimum are found. Thus, the binary-tournament-elitist scheme was chosen for further trials because it retained the most alleles while insuring that the best individual was not lost. A comparison of 100 and 200 population sizes, as well as single-point crossover and uniform crossover with creep mutations, was then made (Fig. 7). All four trials found a minimum with a best error of approximately 2%; this indicates three important features. First, population sizes of 100 or 200 are acceptable for this problem. Second, single-point crossover or uniform crossover with creep are acceptable for this problem. Third, the fact that all four trials found different local minimum demonstrates the high degree of multimodalty of this problem.

As the generations progressed, a detailed look at the available alleles (parameter values still remaining) showed that the uniform crossover with creep scheme retained more alleles than the single-point crossover scheme during the search procedure. Therefore, although all four of these trials produced low error individuals, it appears that the uniform crossover with creep scheme is more likely to find a global minimum because it retains more alleles during the search procedure. Thus, the binary-tournament-elitist-uniform crossover-creep mutation scheme will be used for future GA chemical laser modeling problems of this nature. A population size of 100 will be used for fast searches, whereas a population size of 200 should be used for more thorough searches.

The four different parameter sets, which produced local minimum with an error of approximately 2% for five data points, were then used to calculate how well each set agreed with the complete set of 43 data points. All four parameter sets gave reasonable agreement with the data, but the parameter set that gave the best overall agreement with the data had a yield at $\psi = 3$ of 0.53, a water vapor flow rate of 0.08 moles/s, a k_{18} multiplier of 12, $J_{\text{exp}} = 1.1$, and $DCM_{\text{exp}} = 1.0$. Since a yield of 0.53 is within the spread (on the high side) of very scattered data, it is a reasonable value. A water vapor flow rate of 0.08 moles/s is consistent with recent data.[33] The order of magnitude increase in the rate k_{18} is discussed next. The values for J_{exp} and DCM_{exp} are reasonable since the algorithm found the initial a priori assumed values (already discussed) of 1.1 and 1.0, respectively.

The best overall agreement with data was obtained with a k_{18} multiplier of 12. As discussed in Ref. 11, Heaven and Nowlin[34] and Heaven[35] recently made measurements to determine the $I_2^* + He$ deactivation rate k_{18}. Using an I_2 relaxation simulation model provided by Heaven,[35] k_{18} should be in the range of 1.1×10^{-11}–2.3×10^{-11} under typical laser cavity flow conditions (5 torr and 150 K); this is a factor of approximately 3–6 above the original assumed rate of 4×10^{-12}. Although Heaven's data indicate a rate that is about a factor of 2 smaller than the factor of 12 increase deduced in this study, there is definitely agreement that the nominal $I_2^* + He$ deactivation rate is too slow. As a possible explanation of the discrepancy between the results of this study and Heaven's data, it has been

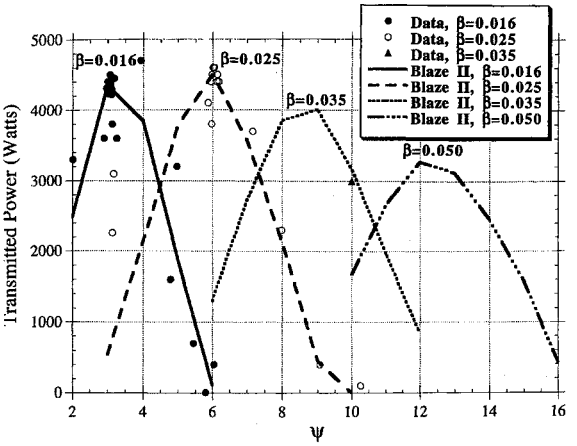

Fig. 8 Transmitted power as a function of ψ for different values of β; Blaze II parameters are a yield of 0.53 at ψ = 3, 0.08, H_2O, 12 × k_{18}, J_{exp} = 1.1, and DCM_{exp} = 1.0.

suggested that the $I_2^* + O_2(^1\Delta) \rightarrow 2I + O_2(^3\Sigma)$ dissociation reaction k_{15} (which has a low-to-medium degree of confidence[26]) may be too fast.[35] If this dissociation reaction were slowed down by roughly a factor of 2 in conjunction with a factor of 6 increase in the $I_2^* + He$ deactivation rate, the effects may be similar to those of increasing the $I_2^* + He$ deactivation rate by a factor of 12.

Figure 8 shows excellent agreement between data and Blaze II with the 0.53/0.08/12/1.1/1.0 parameter set as a function of ψ for three different values of β. The model not only predicts the power very well but also the upward and downward trends. These mixing COIL calculations for the RADICL nozzle indicate that higher iodine flow rates are necessary to maintain a significant fraction of the nominal performance as the total pressure is incresed by the addition of helium; this agrees with RADICL experimental data. Regardless of any minor discrepancies, Fig. 8 suggests that Blaze II can now be used with some degree of confidence to make realistic qualitative predictions of COIL performance trends for the RADICL nozzle. While Blaze II predictions within the flow regimes studied here can be considered quantitative, predictions at considerably different conditions (for example, ψ = 15–20) should be considered qualitative until data exist to verify the model under such conditions. At very high pressures of around 300 torr or more (approximately $\psi > 30$) turbulence may become an issue[11]; this could seriously impact the modeling.

IV. Concluding Remarks

At this point it is possible to answer the question of the utility of the GA technique. First, the genetic algorithm found a parameter set that was a better match than any that were determined by earlier trial-and-error calculations. Second, the GA found this parameter set in 8 days of running on an IBM RS/6000 machine, which is faster than the 2 weeks of personal effort spent doing trial-and-error calculations. In fact, since the GA was performing the calculations automatically, this allowed the user to address other problems during this time period. The binary coding, tournament selection, uniform crossover, jump and creep mutation scheme with elitism appears to work the best for this application. The floating point-tournament-creep-elitist and binary-tournament-elitist schemes (both using single-point crossover) also worked very well. Although several different GA schemes have been compared in this study, it is suggested that future use of the GA for this type of chemical laser modeling problem should include an investigation of the statistical effects of different random number seeds as a function of the GA scheme. Overall, the GA technique worked exceptionally well for this chemical laser modeling problem in a cost effective and time efficient manner. Because of a GAs population-based approach it is inherently parallel; therefore, further modifications could enhance this technique to be extremely fast at searching a large parameter space on a parallel processing computer.

This is the first known application of the genetic algorithm technique for modeling lasers, chemically reacting flows, and chemical lasers. Blaze II was baselined to existing RADICL power data with very good agreement. If a sufficient amount of gain data are available for a broad range of flow conditions, it should be possible to apply this GA technique for baselining to zero power gain data (at the time of this modeling study there was a lack of RADICL gain data covering a broad range of flow conditions). Mixing COIL calculations for the RADICL nozzle indicate that higher iodine flow rates are necessary to maintain a significant fraction of the nominal performance as the total pressure is increased by the addition of helium; this agrees with RADICL experimental power data. Now that the unknown parameters for the RADICL device are established and Blaze II is baselined to experimental data, it may be possible to implement this genetic algorithm technique to optimize the COIL performance as a function of any of the flow rates, mirror location, mirror size, nozzle configuration, injector sizes, and other possible factors. It may also be possible to use this modeling technique for other types of chemical lasers, e.g., hydrogen–fluoride (HF) devices. This modeling procedure could be used as a method to guide experiments to enhance chemical laser performance.

Acknowledgments

This work was supported by the Air Force under Contract F33615-89-C-2912. Many thanks to J. Shaw for providing the RADICL data, to D. Goldberg for many useful genetic algorithm tips, and to G. Hager, D. Plummer, L. Sentman, W. Solomon, and T. Madden, all for numerous useful conversations.

References

[1]Truesdell, K. A., Lamberson, S. E., and Hager, G. D., "Phillips Laboratory COIL Technology Overview," AIAA Paper 92-3003, 1992.

[2]McDermott, W. E., "Singlet Delta Generator Performance—Overview," AIAA Paper 92-3005, 1992.

[3]McDermott, W. E., "The Generation of Singlet Delta Oxygen: A Technology Overview," *Intense Laser Beams and Applications*, Vol. 1871, International Society for Optical Engineering, Los Angeles, CA, 1993, pp. 135–147.

[4]Thayer, W. J., III., "Sensitivity of the Uniform-Droplet Oxygen-Generator Output to Flow Conditions and Geometry," *Intense Laser Beams and Applications*, Vol. 1871, International Society for Optical Engineering, Los Angeles, CA, 1993, pp. 193–202.

[5]Copeland, D. A., Quan, V., Blauer, J. A., and Rodriquez, S. E., "Two-Phase Model of $O_2(^1\Delta)$ Production with Application to Rotating Disk Generators," *Intense Laser Beams and Applications*, Vol. 1871, International Society for Optical Engineering, Los Angeles, CA, 1993, pp. 203–228.

[6]Whitefield, P. D., Hagen, D. E., Trueblood, M. B., Barnett, W. M., and Helms, C., "Experimental Investigation of Homogeneous and Heterogeneous Nucleation Condensation Processes and Products in COIL," *Intense Laser Beams and Applications*, Vol. 1871, International Society for Optical Engineering, Los Angeles, CA, 1993, pp. 277–288.

[7]McDermott, W. E., "The Generation of Singlet Delta Oxygen—A Technology Overview," AIAA Paper 93-3220, 1993.

[8]Copeland, D. A., McDermott, W. E., Quan, V., and Bauer, A. H., "Exact and Approximate Solutions of the Utilization and Yield Equations for O2(1D) Generators," AIAA Paper 93-3220, 1993.

[9]Nowlin, M. L., and Heaven, M. C., "Spectroscopy and Energy Transfer Dynamics of I_2 Levels in the 1.0–1.3 eV Range," *Intense Laser Beams and Applications*, Vol. 1871, International Society for Optical Engineering, Los Angeles, CA, 1993, pp. 290–303.

[10]Crowell, P. G., and Plummer, D. N., "Simplified Chemical Oxygen Iodine Laser (COIL) System Model," *Intense Laser Beams and Applications*, Vol. 1871, International Society for Optical Engineering, Los Angeles, CA, 1993, pp. 148–180.

[11]Carroll, D. L., "Modeling High-Pressure Chemical Oxygen-Iodine Lasers," *AIAA Journal*, Vol. 33, No. 8, 1995, pp. 1454–1462.

[12]Sentman, L. H., Subbiah, M., and Zelazny, S. W., "Blaze II: A Chemical Laser Simulation Computer Program," Bell Aerospace Textron, TR H-CR-77-8, Buffalo, NY, Feb. 1977.

[13]Holland, J. H., *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge, MA, 1992.

[14]Goldberg, D. E., *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison–Wesley, Reading, MA, 1989, Chaps. 1 and 4, pp. 77, 106–122.

[15]Crowell, P. G., "RECOIL: A One-Dimensional Chemical Oxygen Iodine Laser Performance Model: Part I—Theory," RDA Letter 87-A/K-3-02-1079, Nov. 15, 1989.

[16]Plummer, D. N., private communication, Logicon/RDA, Albuquerque, NM, Aug. 1993.

[17]Hager, G., private communication, Phillips Lab., Kirtland AFB, NM, Aug. 1993.

[18]Shaw, J., private communication, Phillips Lab., Kirtland AFB, NM, Aug. 1993.

[19]Crist, S., Sherman, P. M., and Glass, D. R., "Study of Highly Underexpanded Sonic Jet," *AIAA Journal*, Vol. 4, No. 1, 1966, pp. 68–71.

[20]Driscoll, R. J., "Mixing Enhancement in Chemical Lasers, Part I: Experiments," *AIAA Journal*, Vol. 24, No. 7, 1986, pp. 1120–1126.

[21]Driscoll, R. J., "Mixing Enhancement in Chemical Lasers, Part II: Theory," *AIAA Journal*, Vol. 25, No. 7, 1987, pp. 965–971.

[22]Cohen, L. S., Coulter, L. J., and Egan, W. J., Jr., "Penetration and Mixing of Multiple Gas Jets Subjected to a Cross Flow," *AIAA Journal*, Vol. 9, No. 11, 1971, pp. 718–724.

[23]Fristrom, R. M., and Westenberg, A. A., *Flame Structure*, McGraw–Hill, New York, 1965, Chap. 12.

[24]Sentman, L. H., Tsioulos, G., Bichanich, J., Carrol, D. L., Theodoropoulos, P., Gilmore, J., and Gumus, A., "A Comparative Study of cw HF Chemical Laser Fabry-Perot and Stable Resonator Performance," *Proceedings of the International Conference on Lasers '85*, edited by C. P. Wang, STS Press, McLean, VA, 1986, pp. 281–287.

[25]Sentman, L. H., Nayfeh, M. H., Renzoni, P., King, K., Townsend, S., and Tsioulos, G., "Saturation Effects in a cw HF Chemical Laser," *AIAA Journal*, Vol. 23, No. 9, 1985, pp. 1392–1401.

[26]Perram, G. P., and Hager, G. D., "The Standard Chemical Oxygen-Iodine Laser Kinetics Package," Air Force Weapons Lab., AFWL-TR-88-50, Kirkland AFB, NM, Oct. 1988.

[27]Davis, L., *Genetic Algorithms and Simulated Annealing*, Pitman, London, 1987.

[28]Goldberg, D. E., "A Note on Boltzmann Tournament Selection for Genetic Algorithms and Population-Oriented Simulated Annealing," *Complex Systems*, Vol. 4, Complex Systems Pub., 1990, pp. 445–460.

[29]Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T., *Numerical Recipes: The Art of Scientific Computing (FORTRAN Version)*, Cambridge Univ. Press, Cambridge, England, UK, 1989.

[30]Goldberg, D. E., Deb, K., and Clark, J. H., "Genetic Algorithms, Noise, and the Sizing of Populations," *Complex Systems*, Vol. 6, Complex Systems Pub., 1992, pp. 19, 333–362.

[31]Goldberg, D. E., private communication, Univ. of Illinois at Urbana–Champaign, Urbana, IL, Dec. 1993.

[32]Goldberg, D. E., "Real-coded Genetic Algorithms, Virtual Alphabets, and Blocking," *Complex Systems*, Vol. 5, Complex Systems Pub., 1991, pp. 139–167.

[33]Plummer, D. N., private communication, Logicon/RDA, Albuquerque, NM, Aug., 1994.

[34]Heaven, M. C., and Nowlin, M. L., "The Role of Excited Molecular Iodine in the Chemical Oxygen Iodine Laser," AIAA Paper 94-2430, 1994.

[35]Heaven, M. C., private communication, Emory Univ., Atlanta, GA, July 1994.