ELSEVIER

Fuel Processing Technology 88 (2007) 51 – 63

FUEL PROCESSING TECHNOLOGY

www.elsevier.com/locate/fuproc

Multi-objective optimization of fuel oil blending using the jumping gene adaptation of genetic algorithm

Dinesh K. Khosla [a], Santosh K. Gupta [a,*], Deoki N. Saraf [b]

[a] *Department of Chemical Engineering, Indian Institute of Technology, Kanpur 208016, India*
[b] *Department of Chemical Engineering, University of Petroleum and Energy Studies, Dehradun 248007, India*

Received 25 April 2006; received in revised form 24 August 2006; accepted 29 August 2006

Abstract

Production and marketing of heavy fuel oil (HFO) are an easy, effective and economical way to dispose off certain very heavy refinery streams such as short residue (SR, available from the bottom of vacuum distillation units) and clarified liquid oil (CLO, available from the bottom of the main fractionators of fluidized-bed catalytic crackers). Certain lighter streams such as heavy cycle oil (HCO), light cycle oil (LCO) and kerosene, are added to the heavy residual stock to improve its quality in terms of fluidity, combustibility, etc., to be marketed as fuel oil. The present study aims at optimization of the fuel oil blending process to maximize profit, minimize quality give-away, maximize production, minimize use of lighter products such as LCO and kerosene, and maximize the calorific value, etc. Several multi-objective optimization problems have been formulated comprising of two and three-objective functions and solved using the elitist non-dominated sorting genetic algorithm (NSGA-II). This evolutionary technique produces a set of non-dominating (equally good) Pareto optimal solutions from which the operator can choose the one that is most suitable (preferred point). Also, a fixed-length macro–macro mutation operator, inspired by jumping genes in natural genetics, has been used with NSGA-II to solve this problem. This modified algorithm leads to a significant reduction in the computational effort. Indeed, this adaptation can be of immense use in reducing the computational effort for other problems in chemical engineering.

Keywords: Fuel oil; Blending; Optimization; Genetic algorithm; Multi-objective optimization; Pareto sets; Jumping gene

1. Introduction

Blending of various semi-finished or finished components is a common operation in the petroleum, pharmaceutical, cosmetics and food industries. The blend compositions are so chosen that the products meet market specifications or conform to environmental requirements. Optimization of such operations is generally done for the maximization of profit, optimization of usage and minimization of quality give-away. The classic blending problem arises in petroleum refineries since they almost always sell finished products made from more than one component. Blending of crude oil, gasoline, diesel oil, LPG, lube oil, bitumen and heavy fuel oil (HFO) is carried out in most refineries. The refiner desires to select the best combination of blend components, and often varies the formulation depending on price and availability. Some information on the optimization of in-line and off-line (batch) operations is available in the open literature [1,2]. However, no comprehensive study on the off-line optimization of HFO blend recipes is available. Given the high volume of raw material available for HFO blend production in refineries, the global optimization of the blend recipe could lead to substantial savings in cost, resulting in higher profit margins. This study is concerned with the off-line optimization of HFO blends, motivated, in part, by an industry in India.

HFO (also called bunker C fuel, bunker fuel oil, bunk oil, black liquor, marine fuel oil, residual fuel, Type 6 heating fuel oil, No. 6 fuel oil and furnace oil) is a low grade fuel primarily composed of short residue (SR, obtained as the bottoms or residue from the crude vacuum distillation unit) blended with smaller quantities of distillates, namely, clarified liquid oil (CLO), light cycle oil (LCO), heavy cycle oil (HCO) and

* Corresponding author. Tel.: +91 512 259 7031, +91 512 259 7127; fax: +91 512 259 0104.
E-mail address: skgupta@iitk.ac.in (S.K. Gupta).

0378-3820/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.fuproc.2006.08.009

Table 1
Cost price and calorific value [23,24] of the components used for preparing the HFO blend

Component	Price (Rs./ton)	10^{-3} CV_i (kJ/kg)
Short residue	8000	38.311
Clarified liquid oil	11,000	41.860
Heavy cycle oil	12,000	44.790
Light cycle oil	12,500	45.209
Kerosene	13,500	46.465

kerosene, to meet market requirements or standards of specific customers. The physical properties that describe the grade of HFO produced are: kinematic viscosity (ν), specific gravity (sg), flash point (FP), pour point (PP), calorific value (CV), and the sulfur content (Swt.%) of the blend. The viscosity determines the storage and handling temperature if the pour point of the fuel is low. The temperature for atomization of the fuel also depends on the viscosity. Knowledge of specific gravity is required for quantity calculations and for proper sizing of disc-stack centrifuges for clarification. In addition, the specific gravity gives an indication of other fuel characteristics, including specific energy and ignition quality. Oils of the same specific gravity do not necessarily have the same viscosity. The FP is important for minimizing the risk of fire. The PP is important for ascertaining a safe storage temperature for the fuel since the latter is generally stored at temperatures at least 5 to 7 °C above the pour point. CV is important since it represents the heat released during burning. The presence of sulfur in HFO can give rise to air pollution and corrosion problems.

In a modern refinery, there is a wide range of diluents available for the production of HFO from SR. The optimal use of these diluents to produce a saleable product having the required properties while simultaneously improving the profit is important. Linear programming (LP) [3,4] was probably first used [5] by Symonds [6] to solve a simplified gasoline blending problem. Similar work on gasoline blending was reported by Magoulas et al. [7]. Indeed, LP has been widely accepted in the refining industry as a major planning, scheduling and optimization tool due to its ease of use and faster convergence. Unfortunately, the equations describing the product properties (e.g., viscosity, specific gravity, FP and PP) are non-linear. The use of LP for solving such non-linear problems led to the impairment of the blending results. This led to the use of more complex algorithms like non-linear programming (NLP) [8–10]. Analogous procedures were developed to approximate the non-linearity associated with the blending models, e.g., by characterizing HFO using the viscosity index instead of viscosity. However, these algorithms do not guarantee global optima since linearization could restrict the feasible space. Successive (sequential) linear programming (SLP) [11] was subsequently used in which the objective function and the constraints were linearized, LP was used and this sequence was repeated several times. This technique was often found to converge to local optima. For 'nearly-linear' problems this was found to be adequate [11]. These techniques have been reviewed by Tawarmalani and Sahinidis [12]. Techniques based on sequential quadratic programming (SQP) [13,14] were found to be superior where one solves a non-linear optimization problem by successively solving a series of quadratic programming sub-problems. More robust AI-based optimization techniques are available these days which usually converge to global optima, and are replacing the earlier methods.

The focus in this study is the single and multi-objective optimization of HFO blending operations using adaptations of genetic algorithm (GA) [15]. In case of multi-objective optimization, a set of non-dominated (equally good) Pareto optimal solutions [16] is often obtained. Pareto sets are such that when one moves from any one point to another on the set, at least one objective function improves while at least one other deteriorates. The *elitist* non-dominated sorting genetic algorithm-II, referred to as NSGA-II [16,17], is quite a popular technique for obtaining the Pareto solutions. This algorithm incorporates the concept of elitism so as to make it more powerful than the earlier version, NSGA-I [16,18]. Both NSGA-I and NSGA-II give the entire Pareto set in a single application of the algorithm. Unlike SQP, initial guesses are not required in these techniques. These algorithms have been used to solve a variety of multi-objective optimization problems in chemical engineering, as reviewed by Bhaskar et al. [19]. An adaptation of NSGA-II, referred to as NSGA-II-aJG, inspired by the concept of jumping genes (JG) or transposons [20] in natural genetics, is presented in this study to explore if faster convergence can be achieved. This is an improved version of the NSGA-II-JG code developed by Kasat and Gupta [21].

2. Formulation

2.1. Model equations

The model equations for the mass balances are simple (and so are not given here). Perfect mixing is assumed. Due to the non-availability of data on various capital and operating costs (e.g., the cost of the blender, piping costs, inventory costs, pumping and maintenance costs, etc., which typically constitute less than about 2–3% of the total cost) associated with blending, these were not included in the cost of the blend. The amortized costs would be constant for an existing blending unit, and would only reduce the profits by a constant amount, without affecting the general trends of the final results. The manufacturing costs per ton of two key components in the blend, SR and CLO, have been assumed (after discussions with the refinery whose

Table 2
Bounds and constraints used in the optimization problems

		Lower	Upper
Availability bounds (ton)	Short residue	0.0	650
	Clarified liquid oil	0.0	150
	Heavy cycle oil	0.0	250
	Light cycle oil	0.0	300
	Kerosene	0.0	100
Blend property bounds	Specific gravity	0.95	1.0
	Flash point (°C)	60	80
	Pour point (°C)	−40	−22
	wt.% sulfur	3.5	4.5
	Viscosity (m^2/s)	6.0×10^{-5}	3.0×10^{-4}