International Journal of Advanced Research in

Civil,Structural,Environmental and Infrastructure Engineering and Developing

Volume: 1 Issue: 3 25-Apr-2014,ISSN_NO: 2320-723X

Review of Optimization Methods for Aggregate Blending

Priyansh Singh[1] and Gurpreet Singh Walia[2]

Research Scholar, Department of Civil Engineering, Indian Institute of Technology Delhi, Hauz Khas, New Delhi- 110016, INDIA[1].
Assistant Professor, Department of Civil Engineering, Amritsar College of Engineering and Technology, Amritsar, INDIA[2]
cez138070@civil.iitd.ernet.in [1], gurpreet.singh.walia.1@gmail.com[2]

ABSTRACT— *The aggregates for asphalt mix has to be selected from various stockpiles to match the specified gradation requirements. The fraction of various aggregates which give the desired aggregate gradation is very important to insure quality mix. Previously this fraction is determined by graphical and trial & error method. But due to present need, mix requires more sizes of aggregate which is not computable from these traditional methods. Many optimization techniques are now available which can be used for aggregate blending. These methods can seamlessly use to optimize the either specification requirement or cost minimization or both simultaneously. Here in this paper more scientific and mathematical optimization approaches are presented which can accurately answer these problems.*

Keywords— Aggregate blending, Proportioning, Asphalt mixes, Optimization, Linear programing, Quadratic Linear Programing, Genetic algorithm etc...

1, INTRODUCTION

Asphalt mix consists of aggregates and binder. The aggregates make the basic skeleton of mix whether the binder acts to hold them together in the matrix. The load in the asphalt mix is dissipated by particle to particle load transfer mechanism, so the aggregate packing is an important constraint for any mix to result in better performance. Assorted sizes of aggregates are used in the asphalt mix so that voids created by bigger aggregates can be used up by smaller one and so on. The proper selection different sizes of aggregates to insure better performing mix is known as aggregate blending. Aggregates also govern the cost of the mix, thus minimization of cost an also be taken account in the blending.

There are several methods available for aggregate blending which can broadly classify into three categories: a) Graphical Methods; b) Trial and Errors and c) Methods which involve Optimization Techniques. The graphical methods are applied for early stage of asphalt construction and still popular among engineers due to its simplicity and rapidity. Even these methods can be applied in the field to quickly assess the proper aggregate proportioning. The several popular graphical methods are Triangular Chart Method [1], Asphalt Institute Method [2] and Routhfutch Method.

Graphical methods are limited by number of aggregate sizes. Asphalt Institute graphical method [2] and the triangular chart method cannot accommodate more than two and three sizes of aggregates respectively. The results obtained from the graphical methods are roughly accurate and cannot be directly used. Although graphical methods can be used as an initial tool for aggregate proportioning and the solution obtained can be further optimized by trial and error method. The use of trial and error method also become complex with increases the number of different sizes of aggregates. Also the trial and error methods and graphical methods cannot be used to optimize cost.

The aggregate blending problem which affects a large number of aggregates and more than one constraint cannot be solved by trial & error or graphical methods. These complex aggregate blending problems require more accurate and mathematical approach for acceptable results. To address these issues several optimization tools are developed by continuous research in the field. This review paper discusses various optimization methods and examines the applicability of these methods for different optimization problem in following section.

2, AGGREGATE BLENDING THROUGH PRINCIPLE OF LEAST SQUARE

Least square principle is a powerful tool for optimization problems. In this method squares of error or deviation is minimized in order to get optimum solution. The residue of errors are adjusted in way that the squares of errors become minimum as shown in Figure 1. In case of blending problem either deviation from specification mean or cost of aggregates can be considered as minimization criteria. Neumann [3] used the least square principal and minimizes the deviation from specification mean, and aggregate cost is minimized by Ritter et al. [4].

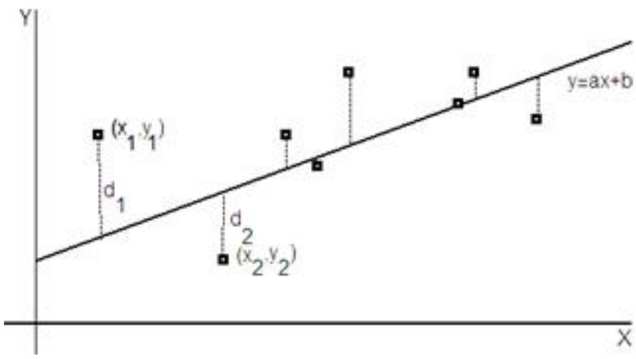

Figure.1 Minimization of residue d_1, d_2 ……d_n in order to get optimum solution

Normally the errors follow the Gaussian curve i.e. the smallest error is committed in large numbers and the larger errors are committed in small numbers as shown in Figure 2.

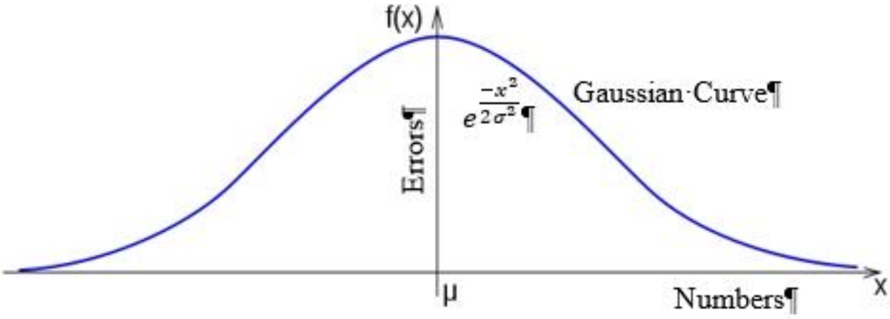

Figure.2 Distribution of errors

K Numbers of aggregates have to be sieved from N numbers of sieves and the percentage passing from each sieve is donated by P_{ij} where $i = 1, 2, 3 \ldots \ldots \ldots . N$ and $j = 1, 2, 3 \ldots \ldots \ldots K$. Upper and lower limit of the specification is denoted by S_{1i} and S_{2i} respectively. $\bar{S}_{mi}$ is mean value of specification limit and S_{mi} is actual total percentage of aggregates passing from for sieve i. Then errors or deviation X_i will be given by $S_{mi} - \bar{S}_{mi}$. The probability that this error X_i lies between X_i and $X_i + dX_i$, where dX_i is confidence interval, is

$$P_i = \frac{h}{\sqrt{\pi}} e^{-h^2 X_i^2}\, dX_i; \;\; where\; h^2 = \frac{1}{2\sigma^2} \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots (1)$$

Hence probability of N errors to occur will be

$$P = \prod_{i=1}^{N} P_i \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots (2)$$

$$P = \left(\frac{\sigma}{\sqrt{\pi}}\right)^N e^{-h\,(\sum_{i=1}^{N} X_i^2)} \prod_{i=1}^{N} dX_i \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots . \ldots (3)$$

Hence the probability P will be maximized when the value of $\sum_{i=1}^{N} X_i^2$ will minimum. This is known as principal of least squares.

If N different aggregates are to be blended to meet the mean specification requirement passing from M number of sieves. The objective function will be as following

$$Minimise, \qquad Z = \sum_{i=1}^{N} X_i^2 = \sum_{i=1}^{N} (S_{mi} - \bar{S}_{mi})^2 \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots . (4)$$

$$Where, \qquad S_{mi} = \sum_{j=1}^{N} P_{ij} \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots . \ldots \ldots (5)$$

This objective function can be solved by treating as unconstrained optimization or constrained optimization. In the case of unconstrained optimization, the function (4) will be resolved without any condition but the accuracy of the solution will be very limited [5]. In order to obtain better results this objective function (4) can be subjected to numbers of constraints such that the percentage passing form each sieve should be within the specification limit i.e.

$$S_{mi} \geq S_{1i}; \ldots \ldots \ldots \ldots (6)$$

$$S_{mi} \leq S_{2i}; \ldots \ldots \ldots \ldots (7)$$

Some more constraint can be imposed which insure Aggregate proportion f_j for all K types of aggregates should be non-negative and their sum should be unified.

$$\sum_{j=1}^{K} f_j = 1; \ldots \ldots \ldots \ldots (8)$$

$$f_j \geq 0; \ldots \ldots \ldots \ldots (9)$$

This system of inequalities and be solved by any solver program. Hence the proportion of each aggregates can be found. In the place of mean deviation one can also optimize this problem for aggregate cost function [6]. The constraints for Bailey Method [7], [8] which applies aggregates ratios on coarse aggregate, fine aggregate and filler in order to control volumetric properties can also applied. Although this method is simple and not required any technical expertise. But the problem will become complex and cannot be treated with simple least square techniques (linear programing) if two or more than two optimization criteria are imposed simultaneously [9], [10]. Although by advancing the constraints and solving methodology this blending problem can be solved for two or more than two optimization criteria. The problem to essentially deal with two or more criteria includes some nonlinear function and can be solved by Nonlinear Programming.

3, AGGREGATE BLENDING BY NON LINEAR PROGRAMING

Nonlinear Programming (NLP) is a mathematical tool which answers the optimization problem which is characterized by nonlinear function [11]. In aggregate blending problem only the objective function is nonlinear and all the constraints are linear. This type of problems is specially recognized as Linear Quadratics Problems (LQP) because they can be solved using numerical techniques that exploit their particular geometry.

If the aggregate blending is to minimize for aggregate cost also, one more cost optimization function is to be used other than function (4). If C_j is the cost of j^{th} aggregate and C is the specified maximum cost for aggregate blend. Then cost optimization function can be expressed as

$$\sum_{j=1}^{K} C_j f_j \leq C \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (10)$$

Now in this problem, first objective is to optimize the deviation from specification mean and the second objective is to minimize the cost of blend. Because both these objective are important, one have to decide priority for these objectives. If first objective has λ priority level and second objective $1-\lambda$ importance. The LQP can be formulated using equation (4) to (10) as

$$Minimise, Z = \lambda \sum_{i=1}^{N} (d_i)^2 + (1-\lambda) N C^2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (11)$$

Where, N is the total numbers of sieve and $d_i = S_{mi} - \bar{S}_{mi}$

The equation (11) and constraints can be expressed in vector matrix format as

$$Minimise, \quad Z = LX + X^T QX \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (12)$$

$$Subject\ to, \quad BX = b; \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (13)$$

$$X \geq 0 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (14)$$

Where, B is a matrix of constraint coefficient; b is column vector; L is a row vector; X vector of decision variable and the matrix Q can be expressed as equation (15). The matrix Q has only non-negative diagonal elements and zero off diagonal element hence the matrix is positive semi definite matrix and this can be solved by Khun-Tucker optimality condition. It can be solved by the simple solution algorithms or any standard solver program. Ravindran [9], [12] worked on the quadratic programming problem and developed algorithm based on Complimentary Pivot Method which is more accurate for solving positive semi definite matrix. The popular computational programs like MATLAB® has inbuilt algorithm to solve these type of problems [11]. The results obtained

$$Q = \begin{array}{c} \\ d_1 \\ \vdots \\ d_n \\ C \\ S_{m1} \\ \vdots \\ S_{mN} \\ f_1 \\ \vdots \\ f_k \end{array} \begin{array}{c} \begin{array}{ccccccccc} d_1 & \ldots & d_n & C & S_{m1} & C & S_{mN} & f_1 & \ldots & f_k \end{array} \\ \begin{bmatrix} \lambda & & & & & & & & & \\ & \ddots & & & & & & & & \\ & & \lambda & & & & & & & \\ & & & (1-\theta)N & & & & & & \\ & & & & 0 & & & & & \\ & & & & & 0 & & & & \\ & & & & & & 0 & & & \\ & & & & & & & \ddots & & \\ & & & & & & & & \ddots & \\ & & & & & & & & & 0 \end{bmatrix} \end{array} \ldots\ldots\ldots (15)$$

for Liner Programing and Quadratic Programming is shown in Figure 3. The curve slightly differs for higher values of λ and almost coincides for lower value of λ. The linear model although consist some errors but due to its simplicity still one can use this.

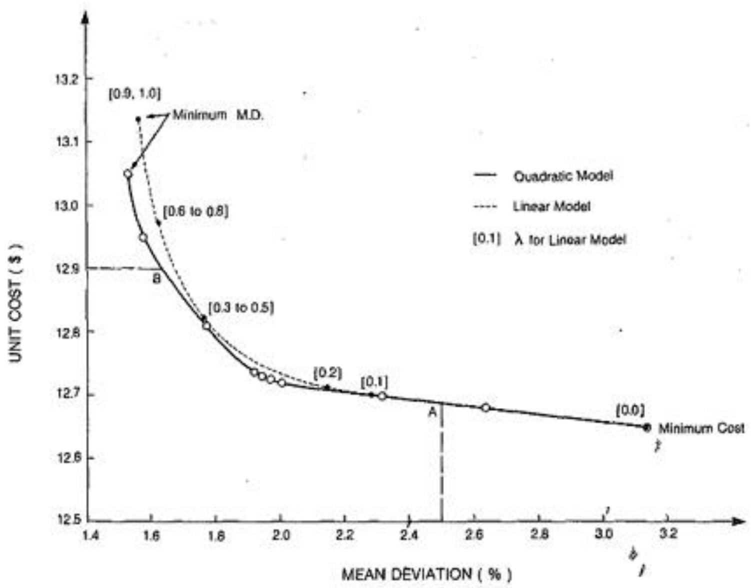

Figure.3 Comparison of results obtained from Linear Programing and λ.

4, AGGREGATE BLENDING BY GENETIC ALGORITHM

Genetic Algorithms (GAs) is evolutionary computation tool for solving multi objective optimization problem. This is a direct search process for determining optimal solution. The algorithms is based on Darwin's theory of "survival of the fittest" [13]. It is developed by Holland in 1975. This method can be very conveniently applied for objectives like cost, given specification etc. GAs can further be generalized to any problem that can arise in the field. The natural activity takes place through the perpetual mutation and recombination of chromosomes in population to yield a better gene structure. The terminology involves with this method is more on the biological side. For the numerical purpose the term chromosome can be treated equivalent to design variable. Fitness is related to the objective function. The population can be evaluated or ranked according to their objective function value.

Algorithm for GAs:

Step 1: Set up an initial population or initial solution $P(0)$.
Evaluate these initial chromosomes for fitness, rate the solution.
Generation index (iteration) $t = 0$
Step 2: Use the genetic operator to generate the set of children (crossover, mutation).
Add new set of randomly generated population (immigrants).
Evaluate the population fitness
Select the members for next generation by competitive selection.

Select population (same numbers for member)
Generation index (iteration) $t + 1$
Evaluate these chromosomes for fitness, if not converged, $t = t + 1$
Again go to Step 2.

The GAs takes the starting set of solutions (population) as an initial generation, and refine this initial generation by and an iterative process. The iteration process involves the genetic operations like reproduction, crossover, and mutation. The solution after every Iteration is taken as initial generation and procedure continues until the required conditions get satisfied [14], [15]

The objective function of GAs can be formulated as equation (4) which is subjected constraint (6), (7), (8) and (9). The resultant gradation curve can be obtained as

$$S_i = \sum_{j=1}^{K} P_{ij} f_j \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (16)$$

Cost optimization criteria can also be applied by considering cost optimization condition (10).

In present problem the vector f_j is taken as chromosomes. The first $K{-}1$ member of chromosome set is randomly selected in the interval [0, 1] and the K^{th} member is calculated as

$$f_k = 1 - \sum_{j=1}^{K-1} f_j \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (17)$$

If the calculated f_k violates the constraint (9) then all chromosomes are rejected and the procedure is repeated with new chromosomes until the satisfactory f_k value achieved. The chromosome set obtained from this process is taken for crossover with another set of chromosomes. Due to this crossover the constraint (9) will be broken. Hence the offspring elements are reproduced with their corresponding normalization factor such that the non-negativity condition can be met. These chromosomes can also be considered for mutation with some randomly generated immigrant and again evaluated for non-negativity condition. If the mutant resulted from mutation fails to meet this condition, it is again multiplied by normalization factor. The fitness of the chromosome is determined from the objective function (16). The whole process is reiterated until reasonable result is obtained.

5, SUMMARY AND CONCLUSION

Aggregate is the most important factor of any mixture. The blending of aggregate is responsible for good packing, adequate air voids etc. By controlling the aggregate gradation one can improve the mixture performance such that resistance to rutting and fatigue etc. The aggregate blending can be performed through numbers of existing method ranging from trial and errors to more complex computation techniques. The popular methods of aggregate blending involve graphical method, least square method, linear and nonlinear programming, Stimulated Annealing techniques and genetic algorithm etc. In the present scenario every contractor is interested in the cost effective aggregate blend. The old methods are acceptable for rough use or to provide initial solution. But in order to obtain a cost effective blend with satisfactory specification requirement one needs to accommodate more sizes of aggregates. These many constraint cannot be optimized with traditional method. Hence the more accurate and capable tools which can resolve these problems are reviewed in this paper.

The least square method is basis of every optimization techniques and results in acceptable blend. Although optimization for more than one objective become complex with the least square method. Nonlinear programing answers the complexities of the least square method. Nonlinear programing results a semi positive matrix due to which problem can be optimized by very simple numerical techniques. Although due to the availability of various computer programs these nonlinear programing problems can easily solve without using any techniques. The result obtained from nonlinear technics are more accurate than the least square method.

The genetic algorithms are most versatile methods for the resolution of any optimization problem. It provides the flexibility of accommodating any numbers of objective and constraints. GAs also provide very simple computation generic. GAs also results very good accuracy and robust solution. The aggregate blending can also be solved with GAs effectively.

Many other methods such as random search methods, steep decent method, ant colony optimization method etc. can also be used for aggregate blending problem.

REFERENCE

[1] M. G. Spangler, *Soil Engineering*, 2nd ed. Scranton, Pa: International Textbook Company, 1960, p. 217.

[2] Asphalt Institute, *Mix Design Methods for Asphalt Concrete and Other Hot-Mix Types*, Manual Ser. The Asphalt Institute, 1962.

[3] D. L. Neumann, “Mathematical Method for Blending Aggregates,” *J. Constr. Div. ASCE*, vol. 90, no. 2, pp. 1–13, 1964.

[4] J. B. Ritter and L. R. Shaffer, “Blending Natural Earth Deposits for Least Cost,” *J. Constr. Div. ASCE*, vol. 87, no. 1, pp. 39–61, 1961.

[5] H. Sivilevičius, V. Podvezko, and S. Vakrinienė, “The use of constrained and unconstrained optimization models in gradation design of hot mix asphalt mixture,” *Constr. Build. Mater.*, vol. 25, no. 1, pp. 115–122, Jan. 2011.

[6] K. Awuah-Offei and H. Askari-Nasab, “Aggregate Cost Minimization in Hot-Mix Asphalt Design,” *Journal of Materials in Civil Engineering*, no. May, pp. 554–561, 2010.

[7] W. Vavrik, W. Pine, and S. Carpenter, “Aggregate blending for asphalt mix design: Bailey method,” *Transportation Research Record: Journal of the Transportation Research Board*, no. 02, pp. 146–153, 2002.

[8] V. Aurilio, W. Pine, and P. Lum, "The Bailey Method Achieving Volumetrics and HMA Compactability," *Proceedings of The Annual Conference-Canadian Technical Asphalt Association*, 2005.

[9] S. Easa and E. Can, "Optimization model for aggregate blending," *Journal of construction engineering and management*, vol. I, no. 3, pp. 216–230, 1985.

[10] S. Easa and E. Can, "Stochastic priority model for aggregate blending," *Journal of construction engineering and management*, vol. I, no. 4, 1985.

[11] P. Venkataraman, *Applied Optimization*, 2nd ed. New Jersey: Jhon Wiley & Sons, 2009, p. 204.

[12] A. Ravindran and H. Lee, "Computer Experiments on Qudratic Programing Algorithms," *Eur. J. Oprational Res.*, vol. 8, no. 2, pp. 166–174, 1982.

[13] Wikipedia Contributors, "Charles Darwin," *Wikipedia, The Free Encyclopedia*, 2013. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Charles_Darwin&oldid=579342633. [Accessed: 19-Oct-2013].

[14] Y. Toklu, "Aggregate blending using genetic algorithms," *Computer-Aided Civil and Infrastructure Engineering*, vol. 20, pp. 450–460, 2005.

[15] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. New York: Addison-Wesley, 1989.