

RESEARCH ARTICLE

Multiobjective PID controller design for active suspension system: scalarization approach

O. Tolga Altınöz

Department of Electrical and Electronics Engineering, Ankara University, Turkey
taltinoz@ankara.edu.tr

ARTICLE INFO

Article History:

Received 06 October 2016

Accepted 14 October 2017

Available 22 April 2018

Keywords:

PID controller

Multiobjective

Scalarization

Particle swarm optimization

Differential evolution

Partitioned method

AMS Classification 2010:

90C26, 80M50, 78M50

49N05, 93C05

ABSTRACT

In this study, the PID tuning method (controller design scheme) is proposed for a linear quarter model of active suspension system installed on the vehicles. The PID tuning scheme is considered as a multiobjective problem which is solved by converting this multiobjective problem into single objective problem with the aid of scalarization approaches. In the study, three different scalarization approaches are used and compared to each other. These approaches are called linear scalarization (weighted sum), epsilon-constraint and Bensons methods. The objectives of multiobjective optimization are selected from the time-domain properties of the transient response of the system which are overshoot, rise time, peak time and error (in total there are four objectives). The aim of each objective is to minimize the corresponding property of the time response of the system. First, these four objective is applied to the scalarization functions and then single objective problem is obtained. Finally, these single objective problems are solved with the aid of heuristic optimization algorithms. For this purpose, four optimization algorithms are selected, which are called Particle Swarm Optimization, Differential Evolution, Firefly, and Cultural Algorithms. In total, twelve implementations are evaluated with the same number of iterations. In this study, the aim is to compare the scalarization approaches and optimization algorithm on active suspension control problem. The performance of the corresponding cases (implementations) are numerically and graphically demonstrated on transient responses of the system.



1. Introduction

Car suspension system has been installed to the vehicle for omitting the undesired low-frequency high magnitude disturbance/vibration due to the imperfect conditions of the road. The main aims of the suspension system are to the comfort of the passengers and absorb the vibrations from ground to car body. Suspension systems can be categorized as depended and independent systems with respect to the connection between suspension and car body. Depended systems are relatively heavy structures which are generally considered as the rear suspension system. Left and right suspensions are (not directly) connected to

each other. Hence change at the dynamics of one of the suspension has a direct effect on the other suspension, which causes discomfort to the passengers. However, since the number of moving parts are less than independent systems, the lifetime of these systems are longer than independent systems under the same conditions. Independent systems are discrete systems. Modern cars prefer independent suspensions, especially for the front wheels.

Since there are many different road conditions, it isn't possible to mention about single vibration frequency and magnitude. Therefore, a particular structure to absorb occurred vibration is needed. There are three main structures can be proposed;

which are passive, semi-active and active suspension systems. Passive systems are pre-design systems where the conditions (relatively protected environment) of the road are known. Therefore, well-designed damper and spring can be answer the suspension problem. They do not have to interconnected devices such as actuator, pneumatics, and hydraulics related to suspension system. Semi-active has at least a damper with a narrow range of variable damping force and damping coefficients. They can be changed before the journey, they don't have any control actions (may be a very simple one). Active systems are improved version of the passive systems with controllable actuators, which can be activated with current and/or voltage. In this paper, active independent suspension system is considered and quarter-car model is evaluated for this purpose.

The control action of the active suspension can be produced vary from classical control approaches, robust control, optimal control, to intelligent methods. Among them, PID controller is the simplest and relatively cheaper controller which is applied to this problem. However, even today the tuning of PID controller is considered as an important aspect. Astrom and Hagglund [1] discussed the future of PID controller in their paper. It is imposed from the study that since there are many aspects which are needed to be considered for tuning, there is a need for design frameworks (section 4). In this study, the PID coefficients for the active suspension control problem is tuned by converting into the multiobjective optimization problem.

In literature, there are some attempts for multiobjective PID tuning problem. In the study of [2], the authors proposed a niche-based multiobjective optimization algorithm and Genetic Algorithm (GA) for tuning of PID controller which is applied to nonlinear MIMO process. As the objective parameters, settling time, overshoot and rise time are weighted and summed. In other words, the authors of that study prefer weighted sum scalarization method. The results showed that proposed method presents better performance when compared with Ziegler-Nichols. Similar results are obtained by Neath et al. [3]. They applied GA-based PID tuning methodology with the aid of weighted sum scalarization method (rise time, settling time and overshoot). In the paper, both simulation and implementation results are demonstrated, and both showed the performance of the tuning method. In [4], Lin et al. applied similar GA-based multiobjective PID design ([5]) method with the aid of weighted sum scalarization. As a difference three time-domain

specifications are considered as objectives which are the rise time, overshoot and steady-state error. A more general contribution of the multiobjective PID tuning is presented by Reynoso-Meza et. al. [6]. Both PID and state space feedback controllers are tuned by using multiobjective optimization algorithm. However, instead of time-domain properties the integral of the absolute value of the derivative the control signal and integral of the absolute value of the error is considered as objectives. This idea is applied to twin rotor MIMO system (as a PI controller tuning algorithm similar study evaluated on [7] and for aircraft in [8]) and results also support the better performance of the multiobjective tuning methodology. Also in [9], two similar objectives which are integral time absolute error and control effort is taken to tune PID controller for the plastic injection molding process, and in [10] same objectives are evaluated on weighted sum scalarization function. Artificial Bee Colony algorithm and weighted sum approach was also applied to load frequency control problem [11], and almost same improvement was reached. Hung et. al. [12] presents the same methodology, but on the contrary the authors preferred multiobjective simulated annealing algorithm and improved strength Pareto algorithms. The other difference is the selection of objective function. In that study, three objectives are considered which are robust stability, disturbance attenuation and integral of square error. Simulation results demonstrate the high performance of the proposed framework. In the study of Tseng et al. [13], the sufficient performance of the multiobjective PID control design in plant uncertainties and under external disturbance, also parametric uncertainties are considered in [14]. In [15], Tang et al. gives the multiobjective optimization scheme (multiobjective generic algorithm (MOGA)) for Fuzzy PID controller. Liu and Daley [16] proposed a three-layered study, in a way that first time-domain optimal tuning of PID control is presented. Following that consequently, frequency-domain optimal-tuning PID and multiobjective tuning PID controllers are discussed. These PID tuning algorithms are applied to three industrial systems. Results showed that optimal PID significantly improve performance. In Ayala and Coelho paper [17], a multiobjective optimization algorithm (NSGA-II) is applied without using scalarization functions (the similar study is presented in [18]). The objectives are selected as position error and torque.

In active suspension problem, the aim is to improve the transient response of the system under

change in the road profile. At the steady state of the system, the damper (or spring) in the system converges/remains in a stable state and this state is not changed if the road remains the same. However, if the road is altered due to the imperfect conditions of the road the suspension system response to the change at the road. If this change is relatively small and the mass differences between body and suspension system including the spring property of the tire is large that this change cannot be perceived. However, if this change is large enough, then the car body moves upwards, and oscillation begins. If this oscillation could not be damped or car body moves too far from the suspension system than safety and comfort of the passengers is reduced dramatically. Therefore a control structure is needed to apply a force on the damper to damp this effect. Hence in this paper, PID controller is applied to the quarter car suspension model. The PID coefficients are the key parameters which are the direct effect on the performance of the system. However, the conventional PID tuning methods could not be applied due to the not satisfactory performance. Hence, in this study, a methodology for PID tuning for active suspension system is proposed. Initially, the tuning scheme of active suspension control is converted into the multiobjective problem. However, since it is desired to get a single solution, instead of multiobjective optimization algorithms, scalarization approaches are evaluated for this purpose. Three scalarization methods are applied in this paper which are Weighted sum method, Epsilon method, and Benson's method. As a result of scalarization, the multiobjective problem is reduced to single-objective one. Then this problem is solved by using heuristic optimization algorithms. Four optimization algorithms are applied to solve the problem, which are Particle Swarm Optimization, Differential Evolution, Firefly Algorithm and Cultural Algorithm. In total, three scalarization approaches with four heuristic optimization algorithm are compared to each other.

This paper is organized into five main sections including the introduction. After the presentation of the aim and literature search at the introduction section, problem definition is given in the second section. The mathematical model of the car suspension system and corresponding controller algorithm with objectives are also presented in this section. The third section is written for briefly explaining the toolsets which are used in this paper. Two sub-sections are given in this section which are scalarization approaches and optimization algorithms. The fourth section is given for implementation and obtained results. In this

section, all the information explained in the previous sections are evaluated on the simulation environment. The last section is the conclusion of this study.

2. Problem definition

The graphical description of the quarter-car passive suspension model [19] is illustrated in Figure 1, and the following equations give the mathematical description of this model [20].

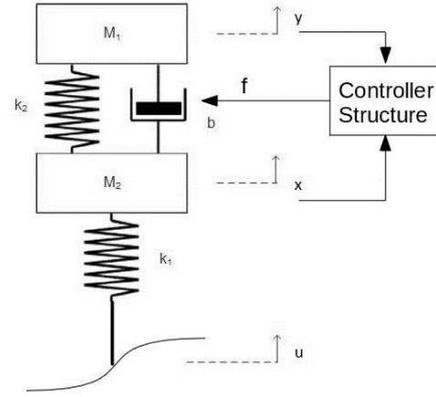


Figure 1. Quarter car suspension system.

$$M_1 \ddot{y} = -b(\dot{y} - \dot{x}) - k_2(y - x) - f + M_2 g \quad (1)$$

$$M_2 \ddot{x} = -b(\dot{x} - \dot{y}) - k_2(x - y) - k_1(x - u) + f + M_1 g \quad (2)$$

where g is the gravitational constant, f is the force (control input) at the damper under the car body, k is the spring constants and b is the damper constant. The masses m_1 and m_2 are corresponded to car body and tire masses. The parameters are selected as: $m_1 = 60\text{kg}$, $m_2 = 300\text{kg}$, $k_1 = 160000\text{kg/s}$, $k_2 = 16000\text{kg/s}$, $g = 9.8\text{m/s}^2$, and $b = 1400\text{kgm/s}^2$.

2.1. PID controller

In this study, PID controllers are designed and parameters are optimized [21]. The Laplace transform of the PID controller is given below.

$$G(s) = K_P + \frac{K_I}{s} + K_D s \quad (3)$$

For proper usage of the PID controller three parameters (proportional parameter (K_P), integral term (K_I), derivative term (K_D)) are needed to

be optimized for the desired performance. As a general perspective, K_P decreases the rise time and steady-state error, but for large steady-state error properly selected K_I is needed to eliminate this large steady-state error. However, this term increases the overshoot. Generally, the increase in overshoot and settling time is relatively small (depends on the structure of the system). If it is large enough, derivative term K_D is added for decreasing the overshoot and settling time. In this study, three values corresponding to these three parameters are determined with the aid of optimization algorithms.

The framework of the controller system is given in Figure 2. The PID controller is applied to the active suspension system, and the road disturbance is changes the system dynamic. By collecting the transient response parameters are converted into objective functions. Next, this objective functions is formed to the multiobjective problem and converted to single objective one with the aid of scalarization approach. Finally, the optimization algorithm is applied to this offline PID parameter tuning framework.

3. Methods and techniques

The study begins with the conversion of the multiobjective problem which is defined in the previous section (Problem Definition) into single objective problem. For this purpose three different scalarization methods are applied to the multiobjective problem. The scalarization methods are called i) weighted sum, ii) ϵ -constrained, and iii) Benson's methods. After conversion to the single-objective problem, two different optimization algorithms are applied to solve this problem. The algorithms are called Particle Swarm Optimization and Differential Evolution algorithms. At the next section, the results of these implementations are compared with each other.

3.1. Scalarization approaches

3.1.1. Weighted sum method

Weighted (linear) sum method is the oldest and best known approach for solving multiobjective optimization problems [22]. The scalarization formula of weighted sum method is given at the following equation.

$$J = \sum_{m=1}^M (w_m f_m(x)) \quad (4)$$

where w are the positive weights of each objective. It is assumed that the sum of all weight are equal to one because of the necessity of normalize values at the objective space.

$$\sum_{m=1}^M (w_m) = 1 \quad (5)$$

The distribution of the solutions obtained from optimization algorithm is generally non-uniform. Also, the weighted sum method can perform better in convex regions of the search space. Since the sum of weights equals to one, w_m parameters are selected as equal to each other, which has the value of $w_m = \frac{1}{m}$. In our study there are four objectives (in the implementation section these objectives are explained). Therefore all of the weights are equal to 0.25.

3.1.2. ϵ -constraint Method

This method is proposed to get rid of the convexity problem of the weighted sum method; ϵ -constrained method was introduced by Haimes et al. (1971) [23]. The idea is based on minimizing the single objective while considering the other objective as constraint in the form of inequality. The scalarization function is presented below.

$$\begin{aligned} J &= f_n(x) \\ f_m(x) &\leq \epsilon, \\ m &= 1, \dots, M; m \neq n \end{aligned} \quad (6)$$

where ϵ is the variable such that with a properly selected ϵ , feasible solution can be obtained. This method can be applied for general problems, no convexity assumption is desired. This scalarization approach converts the unconstrained multiobjective problem into a constrained single objective problem. Therefore, a mechanism is needed to handle these constrained. For our study, there are three constraints are defined for this approach (since we have four objectives). For this purpose penalty function idea is applied to ϵ -constraint method. The penalty function definition is given below.

Definition 1. A function $p(x)$ is said to be penalty function for the vector x if penalty function satisfies two conditions where $g(x) \leq 0$ is the constrained i) $p(x) = 0$ if $g(x) \leq 0$ and ii) $p(x) > 0$ if $g(x) > 0$

For this study, $\sum_{i=1}^m \max(\epsilon_i, f_i(x))$ function is selected as penalty function. As a result the overall

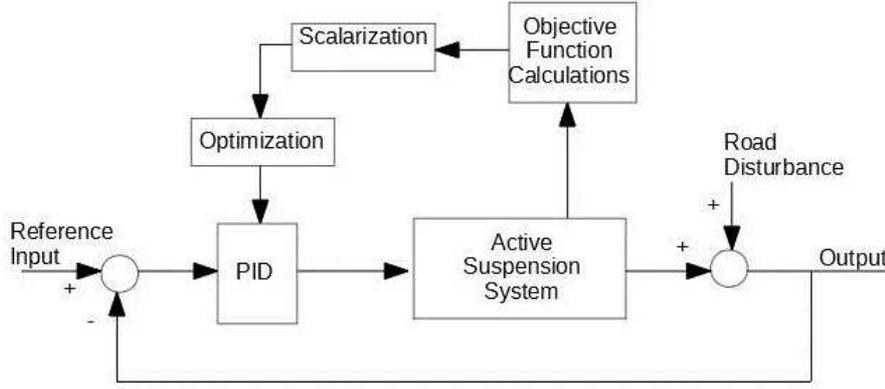


Figure 2. Control structure for quarter car suspension system.

scalarization equation is changed to the following form.

$$J = f_n(x) + \sum_{i=1}^m \max(\epsilon_i, f_i(x)) \quad (7)$$

3.1.3. Benson's method

Benson's method, as one of the scalarization methods, was introduced by Benson in 1978 [24] which is an extended and improved version of their study on vector maximization [25]. In [26], it is shown that the scalarization method is valid in general case without convexity assumption if the reference point is selected properly on the feasible solution set. The idea is based on the determination of the properly or improperly efficient solution, which are defined below.

Definition 2. x_0 is said to be efficient solution if $f_i(x) > f_i(x_0)$ for some x , and there exists at least one j such that $f_j(x) < f_j(x_0)$.

Definition 3. x_0 is said to be properly efficient solution if it is efficient and if there exist a scalar named as $M > 0$, such that for each i , we have

$$\frac{f_i(x) - f_i(x_0)}{f_j(x_0) - f_j(x)} \leq M \quad (8)$$

for some j such that $f_j(x) < f_j(x_0)$ whenever $f_i(x) > f_i(x_0)$.

Definition 4. x_0 is said to be improperly efficient solution if it is efficient and if there exist a scalar named as $M > 0$, such that there is a point x and an i such that $f_i(x) > f_i(x_0)$ and

$$\frac{f_i(x) - f_i(x_0)}{f_j(x_0) - f_j(x)} > M \quad (9)$$

for all j such that $f_j(x) < f_j(x_0)$.

Since the originally proposed method has drawbacks against differentiation, Ehrgott [27] was proposed a modified formulation to make easier of the differentiation process at classical optimization/search algorithms. The scalarization method is given at the following formula.

$$J = \sum_{m=1}^M \max(0, (z_m - f_m(x))) \quad (10)$$

$$f_m(x) \leq z_m, \quad m = 1, \dots, M; m \neq n$$

where z is the chosen solution in the feasible region. First the nonnegative difference between each objective value is calculated and summarized. The maximization is similar to find a cube with the largest perimeter [28]. A set of constraints are added to the formulation. similarly to ϵ -constrained approach, this constraints are handled by using the penalty function.

3.2. Optimization algorithms

In the previous section, scalarization approaches, which are converted multiobjective PID tuning problem into the single objective problem; are explained. In this section, two single objective optimization algorithms are explained, which are Particle Swarm Optimization (PSO) and Differential Evolution (DE).

3.2.1. Particle swarm optimization

Particle Swarm Optimization (PSO) is an optimization algorithm which is proposed by Kennedy and Eberhart [29] in 1995 inspired from the behaviors of the animal swarms. Figure 3 graphically illustrates the idea of the PSO algorithm.

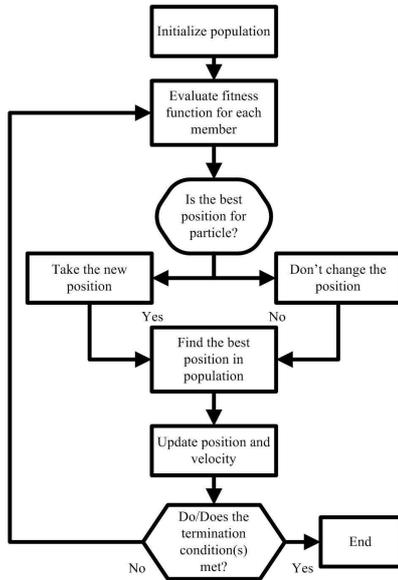


Figure 3. Flow-diagram of particle swarm optimization.

Each member of the population has two properties; position (x_1, x_2, \dots, x_D) and velocity (v_1, v_2, \dots, v_D) , where D is the dimension of the search space. At the beginning of the algorithm, positions are randomly assigned inside the borders of the search space, and similarly velocities are taken value inside $[0, 1]$. At the first phase of each iteration, objective values of each member in population is calculated. There are two kinds of memory defined for each member. The first one is the best position among the population (*gbest*). Same position is recorded for all members at each iteration. The second one is belongs to the each member. It stores the best location which has been ever visited by the corresponding member (*pbest*). Therefore, at the second phase of the algorithm these two memories are updated by using the objective values. Then as the last phase of the algorithm the position and velocities are updated by using the equations given below.

$$v_i[k+1] = v_i[k] + c_1 \text{rand}() (pbest_i - x_i[k]) + c_2 \text{rand}() (gbest_i - x_i[k]) \quad (11)$$

$$x_i[k+1] = x_i[k] + v_i[k+1] \quad (12)$$

where $c_1 = c_2 = 2.05$ are the algorithm parameters and *rand* is the random number generator. These steps are repeated until the termination condition is met.

3.2.2. Differential evolution

Differential Evolution (DE) was proposed by Storn and Price in 1995 [30]. Since DE uses operators mutation, crossover and selection; it is considered as a part of evolutionary algorithms. Figure 4 gives the flow diagram of the DE algorithm.

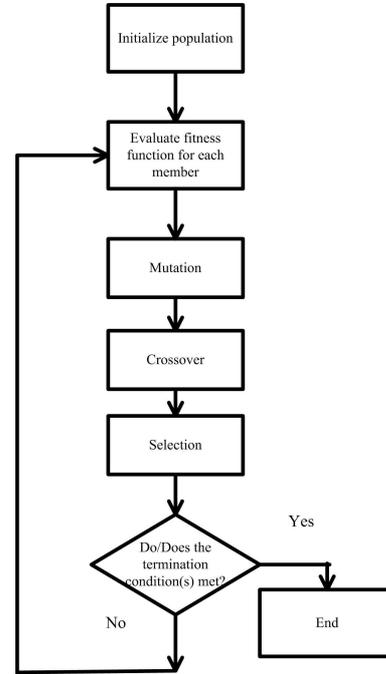


Figure 4. Flow-diagram of differential evolution.

The algorithm begins with the randomly initialization of the population (x_1, x_2, \dots, x_D) on search space, where D is the dimension of the problem. After the initialization of the population is completed, then as the first operator, Mutation, is evaluated. The idea of the mutation operator is to form a new population (v_1, v_2, \dots, v_D) . One of the possible mutation operation (also used in this study) is given below.

$$v_i[k+1] = x_{r_1}[k] + F(x_{r_2}[k] - x_{r_3}[k]) \quad (13)$$

where r_1, r_2 , and r_3 are the randomly selected index from the population, and F is the algorithm coefficient which is selected as $F = 0.8$.

After the mutation operator is completed, then Crossover operation is evaluated. By using this operator a new set of solution is obtained (u_1, u_2, \dots, u_D) , by using the formulation (called binomial operator) given below.

$$u_i[k] = v_i[k] \quad \text{if } \text{rand} < CR \\ u_i[k] = x_i[k] \quad \text{otherwise} \quad (14)$$

where CR is the second algorithm parameter, and it has the value of $CR = 0.9$. The final operator is called the Selection. This operator is just compared the two vector set X and U , and best of these two sets are selected with respect to the objective value. This new set survives to the next iteration.

3.2.3. Firefly algorithm

As a population based optimization algorithm, Firefly Algorithm (FFA) was proposed by Yang [31], [32] in 2008. The algorithm is designed based on the light intensity and attractiveness between fireflies in their nest. Light intensity of fireflies gives the warning ability against predators and attractiveness for mating. The light intensity is changed with respect to the distance between light source and fireflies. The algorithm is designed based on light intensity property, where Figure 5 gives the flow diagram of the FFA algorithm

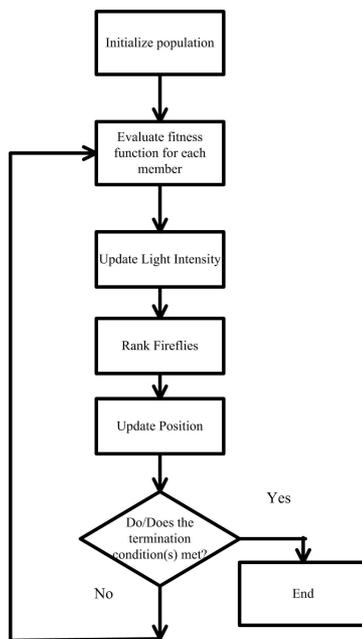


Figure 5. Flow-diagram of firefly algorithm.

In the FFA perspective, the search space is corresponding to the light distribution. Therefore, objective value becomes the light distribution. The performance of the FFA is evaluated on benchmark problems in [33]. The results showed that FFA presents acceptable performance. Then the algorithm is improved for multimodal problems [34] and applied to real-world problems [32], [35]. Fireflies in the algorithm assumes to have position (x_1, x_2, \dots, x_D) on search space. Then, light intensity is calculated by using the position of the member and the distance to other fireflies as formulated in below.

$$I(r) = I_0 e^{-\gamma r^2} \quad (15)$$

where γ is the constant light absorption coefficient, I_0 is the initial light intensity at the distance r , which is defined below.

$$r_{ij} = \|x_i - x_j\| \quad (16)$$

As the last operator of the algorithm the position of each firefly is changed by using the light intensity (attractiveness) and a random movement. Even the algorithm is well organized and proposed acceptable performance, the number of algorithmic control parameters is relatively many in number. Therefore, in this paper the parameters reported in [31] is preferred.

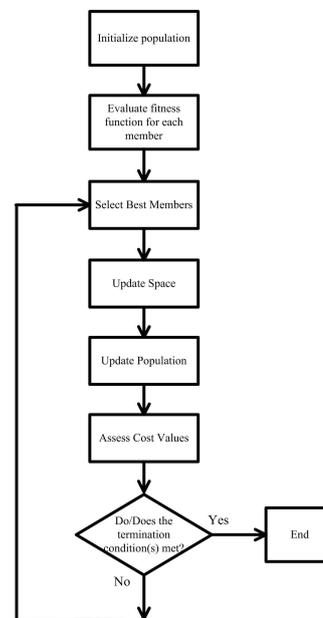


Figure 6. Flow-diagram of cultural algorithm.

3.2.4. Cultural algorithm

Cultural Algorithm (CA) is an optimization algorithm based on social learning and evolution. It was introduced by Reynold [36] and matured in [37]. The algorithm is successfully applied to industrial problems [38]. In CA, the population shares the information pool (in other words belief space). This space contains normative [39], spatial, temporal [40], domain and exemplar knowledge [37]. In interaction of these knowledge is the source of the algorithm [41]. Figure 6 gives the flow diagram of the CA algorithm

In CA, two spaces named as Belief Space and Population space are interacted with each other. The

members in population space are used to calculate the fitness (cost) functions. Based on the cost values the individuals are selected to impact (update) the Belief Space. Then beliefs in the Belief Space have influence the evolution (generation) of the populations in population space. New members are generator with the aid of Belief space and the best members among the joint space remains for the next generator.

4. Implementations and results

The purpose of this study is to design PID controller for active suspension control under the road change. In addition to this purpose, scalarization and heuristic optimization algorithms are compared with each other. For these purposes, first it is assumed that the road change is the step, in other words the disturbance of the system is the step input. All of the implementations are compared with each other with respect to the time response transient properties numerically. The results are discussed and one of from each scalarization methods and optimization algorithms are selected. Then, as the second phase of the study, different road change as ramp input is applied the same system and the performance of the proposed method is discussed with a ramp input, which is applied different ramps.

Initially, the car suspension is controlled under a rapid change at the contact between road and the wheel. It is assumed that there is $1m$ change at this contact, in other words, a step input is applied to the system. It is desired that the suspension is rapidly and smoothly absorb the vibration and return its equilibrium point. First, the problem is defined as a multiobjective problem. The objectives are selected from the time-response of the system. For using at objectives; overshoot, rise time, peak time and error are taken as variables.

Time domain response of the system is divided into two part. The initial part is called the transient response and the rest of it is the steady-state response. For a steady state response the system output (car body (M_1) position change) must be settled within %2 of the desired output which is selected as 1 for this study. Since the mechanical properties of the suspension system is based on damper and springs, it is expected to settle on a certain level. Therefore, even the steady state error is important as an objective, it is not necessary to add as a comparison factor.

The time where the output reaches the desired output (steady state level) for the first time is

called rise time (tr), the time where the response reaches the peak is called peak time (tp). The percentage of the maximum value with respect to the desired response is called overshoot (OS). In addition to these time properties, also error (e) (difference between desired level and the output) is evaluated as the objective, which is important as an objective (as explained previously). The mean integral of absolute difference between desired signal and the output (mae) is calculated and considered as one of the objective.

Corresponding four objectives (f_i , $i = 1, 2, 3, 4$) are given below;

$$\begin{aligned} f_1 &= \min(tr) & f_2 &= \min(tp) \\ f_3 &= \min(OS/100) & f_4 &= \min(mae(e)) \end{aligned} \quad (17)$$

As the next step, this four objective problem is converted into single objective one. Three scalarization functions are preferred, which are weighted-sum, ϵ -constrained, and Benson's methods. In weighted sum method, all of these objectives are summed to each other since all of the weights are same with each other. The single objective of the weighted sum method is given below.

$$f = \frac{1}{4}[tr + tp + (OS/100) + mae(e)] \quad (18)$$

Like weighted sum method, ϵ -constrained method also evaluated for the problem. However, since this scalarization approach is considered, the constrained part of the approach should be evaluated. For this purpose, penalty function is preferred and constrained problem converted into unconstrained problem. The final ϵ -constrained approach formulation is presented below.

$$\begin{aligned} f &= mae(e) + [\max(\epsilon_1, tr) \\ &+ \max(\epsilon_2, tp) + \max(\epsilon_3, OS/100)] \end{aligned} \quad (19)$$

where $\epsilon_1 = 0.05$, $\epsilon_2 = 0.05$, and $\epsilon_3 = 0.2$ are selected. In other words, (as an example) if the overshoot is decreased under 20%, it is considered as the desired performance is reached and the contribution of this property is becomes zero. However, if it is larger than 20%, only the overshoot value is added to the objective value. If all of the given values are above the given ϵ values, than ϵ -constrained method is almost the same as weighted sum method. Similarly, the overall function for Benson's method is given below.

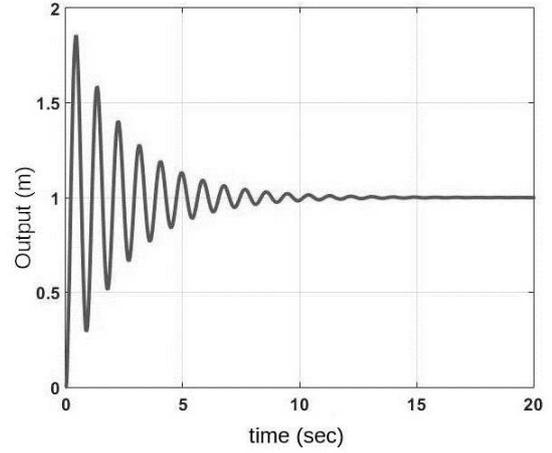
Table 1. Time response parameter comparison with respect to the step input.

Time Response	PSO-WS	PSO-Eps	PSO-Benson	DE-WS	DE-Eps	DE-Benson
Rise Time	0.1637	0.1624	0.1624	0.1366	0.1368	0.1368
Overshoot (%)	48.03	49.99	49.99	50.04	50.04	50.49
Settling Time	0.4389	0.4585	0.4392	0.3542	0.3542	0.3543
	FFA-WS	FFA-Eps	FFA-Benson	CA-WS	CA-Eps	CA-Benson
Rise Time	0.1376	0.1367	0.1365	0.0458	0.0796	0.0796
Overshoot (%)	49.29	49.58	50	10.81	0.0431	0.0431
Settling Time	0.3539	0.3541	0.3541	0.1	0.1224	0.1224

$$\begin{aligned}
f = & \max(0, (z_1 - mae(e))) + \max(0, (z_2 - tr)) \\
& + \max(0, (z_3 - tp)) + \max(0, (z_4 - OS/100)) \\
& + [\max(z_1, mae(e)) + [\max(z_2, tr) \\
& + \max(z_3, tp) + \max(z_4, OS/100)](20)
\end{aligned}$$

where $z_1 = 10$, $z_2 = 0.5$, $z_3 = 0.5$, and $z_4 = 0.2$ are selected. Even this method looks different from other two scalarization functions, it is almost the united version of weighted sum and ϵ -constrained methods such that last objective (mae) also considered inside the maximum function and difference between a reference point and objective is calculated. This scalarization equation can be considered as two parts. In the first part if the obtained property is larger than the z then first part ($\max(0, (z - P))$) become zero. However the second part (the constrained part) is equals to the properties value. On contrary, if z is larger that obtained property, than first part equals to the difference between ($z - P$) and the second part equals to z . When compared to the ϵ -constrained method, a residue from ($\max(0, (z - P))$) is added, which increases when the undesired response obtained.

Since the change at the relative position between ground and car body forms an undesired vibration on the car body. This vibration continues at a certain time if any control action didn't apply. Figure 7 gives the obtained uncontrolled signal. This figure also demonstrates the necessity of the control action. In general, the fastest and low (preferably zero) overshoot is desired. In real world application it corresponds to that after the tire falls into the hole on the road, it is expected to return the car body to its initial height as fast as possible. In addition, it is not desired to move the car body to a higher height of the initial position.

**Figure 7.** Change the relative position of the car body without control action.

From this figure, the car body travels almost $\times 2$ (100 %) more than desired level, and it needs almost 15sec to absorb and reach to the equilibrium point. Also from the figure the number of cycles is more than 10. This figure graphically demonstrates the necessity of the control algorithm for a better drive characteristics. As the final step, the tuned PID controllers are applied to this problem and tried to get a better response from figure 7.

In this study, at first, three scalarization functions are evaluated on four optimization algorithms on car suspension problem which is explained in section 2. All of the optimization algorithms are run 100 iterations with 100 members, also a termination condition is defined to be sure that all of the optimization algorithms calculate the same number of functions evaluations. Table 1 presents the transient response parameters numerically, and the PID parameters are also reported in Table 2.

The results in Table 1 shows that, PSO, DE and FFA are both presents almost the same performance for different PID parameters (Table 2).

Table 2. PID controller parameters, where WS: Weighted Sum, and Eps: Epsilon.

PID Parameters	PSO-WS	PSO-Eps	PSO-Benson	DE-WS	DE-Eps	DE-Benson
KP	696	700	701	988	988	987
KI	266	18.85	206	10^{-3}	10^{-3}	498
KD	927	942	942	978	978	970
	FFA-WS	FFA-Eps	FFA-Benson	CA-WS	CA-Eps	CA-Benson
KP	743	995	999	141	439	141
KI	10^{-3}	1.306	748	159	160	159
KD	989	998	996	716	131	716

However, CA gives the best performance with respect to the transient performance among all heuristic algorithms. It can be concluded that the meta-heuristics of PSO, DE and FFA converges to local optimum, and remains in that solution. It should be also noted that some mechanisms may help to move that from local solution. CA gives the best performance among all other algorithms such that overshoot is almost 10% which is the lowest level and presents fastest rise and settling times. When the results are discussed with respect to the scalarization approaches, it is clear to see that three scalarization approaches are almost similar to each other; however, weighted sum method can able to give the best result in overall. But when compared inside CA algorithm epsilon and Benson's methods give the best results. Even there is a very slight difference between two scalarization methods, it can be seen that the Benson's approach perform better than ϵ -constrained approach.

Table 2 gives the PID parameters obtained from optimization algorithms. In general, PSO, DE, FFA and CE gives almost same parameters for scalarization approaches. For PSO algorithm, K_P and K_D are almost same with each other. DE algorithm searches the solution especially at the border of the search space such that parameters are from lower and higher border values. Like PSO, FFA and CA give same values for K_D and K_I parameters, respectively.

As the second half of the implementation, the optimized PID controller is applied to the system under ramp input. The ramp input corresponds to a slightly change at the level of the road. In real world the road change may happens rapidly, like holes at the road. In addition, speed bump like changes may happen on the road. Therefore, in this part of the study, the performance of the controller under speed bump like changes on the road is investigated. For this purpose, ramp input at different slopes are applied to the problem and obtained results are graphically demonstrated in Figure 8.

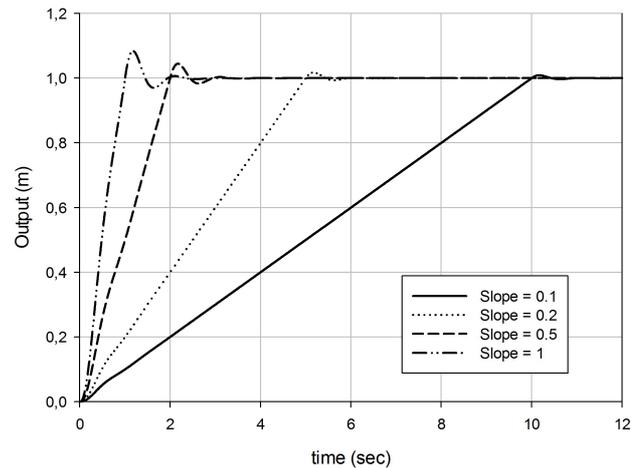
**Figure 8.** Car body position change for ramp input with different slopes.

Figure 8 gives the transient response of the CA-optimized PID controller for different slope ramp input. The figure shows that for slopes 0.1, 0.2, 0.5, and the overshoots becomes 8%, 5%, 2%, and 1.8% respectively. As the slope of the ramp increases, the overshoot and settling time decreases. In other words, as the change of level of the road is more slowly, then the response to this change becomes much better. Since the same optimized parameters are preferred for both implementations, the overall solution also supports the performance of the CA-based PID tuning methodology.

5. Conclusion

In this paper the active independent quarter-car suspension problem is solved by using PID controller. The PID parameter tuning is considered as the multiobjective optimization problem. First this problem is converted into single objective problem by using three scalarization approaches which are weighted sum, ϵ -constrained and Benson's methods. Then the obtained single objective problem is solved with four optimization algorithms which are called Particle Swarm Optimization (PSO), Differential Evolution (DE), Firefly Algorithm and Cultural Algorithm.

There are three aims of this study: i) propose a multiobjective PID tuning scheme for active independent car suspension system ii) compare three scalarization functions and show the implementation of these approaches and iii) compare four optimization algorithms. In total twelve implementations are made and compared with each other. The results showed that even the scalarization approaches present similar performances, among all optimization algorithms, CA gives the best performance. Also, it can be stated that, even a slight difference weighted sum still can be preferred as an efficient scalarization approach.

After CA is selected as the optimizer for PID parameters, a different input as the road change is applied. The ramp input with different slopes are implemented and results are discussed. The results indicate that as the slope of the ramp increases the transient response performance of the overall system is also increases due to the slow change at the level of the road.

In conclusion, CA algorithm presents the best performance among all other optimization algorithms discussed in this paper. The main reason is the local optimum problem of the other algorithms for this problem. Also, scalarization methods can be able to successfully applied to the PID tuning algorithm to convert the multiobjective problem into single objective one. The results indicate that weighted sum presents the best performance overall. The ϵ -constrained and Benson's methods give almost the same performance as PSO, DE, and FFA with a slight difference. As the future study, constrained-based scalarization approaches are deeply investigated with a different set of ϵ and reference points.

References

- [1] Astrom K.J., and Haggglund T., The future of PID control, *Control Engineering Practice*, 9, 11, 1163-1175 (2001).
- [2] Arruda L.V.R, Swiech M.C.S, Delgado M.R.B, and Neves-Jr F., PID control of MIMO process based on rank niching genetic algorithm, *Applied Intelligence*, 29, 3, 290-305 (2008).
- [3] Neath M.J., Swain A.K., Madawala U.K., and Thrimawithana D.J., An Optimal PID Controller for a Bidirectional Inductive Power Transfer System Using Multiobjective Genetic Algorithm, *IEEE Transactions on Power Electronics*, 29, 3, 1523-1531 (2014).
- [4] Lin C.-L., Jan H.-Y., and Shieh N.-C, GA-based multiobjective PID control for a linear brushless DC motor, *IEEE/ASME Transactions on Mechatronics*, 8, 1, 56-65 (2003).
- [5] Herreros A., Baeyens E., and Pern J.R., Design of PID-type controllers using multiobjective genetic algorithms, *ISA Transactions*, 41, 4, 457-472 (2002).
- [6] Reynoso-Meza G., Garcia-Nieto S., Sanchis J., and Blasco F.X., Controller Tuning by Means of Multi-Objective Optimization Algorithms: A Global Tuning Framework, *IEEE Transactions on Control Systems Technology*, 21, 2, 445-458 (2013).
- [7] Reynoso-Meza G., Sanchis J., Blasco X., and Herrero J.M., Multiobjective evolutionary algorithms for multivariable PI controller design, *Expert Systems with Applications*, 39, 9, 7895-7907 (2012).
- [8] Zhaoa S.-Z, Willjuice Iruthayarajanb M., Baskarc S., and Suganthan P.N., Multi-objective robust PID controller tuning using two lbests multi-objective particle swarm optimization, *Information Sciences*, 181, 16, 3323-3335 (2011).
- [9] C.M. Seaman, A.A. Desrochers, and G.F. List, Multiobjective optimization of a plastic injection molding process, *IEEE Transactions on Control Systems Technology*, 2, 3, 157-168 (1994).
- [10] Gambier A., MPC and PID control based on Multi-Objective Optimization, *American Control Conference*, 4727-4732 (2008).
- [11] Naidua K., Mokhlisa H., and Bakar A.H.A., Multiobjective optimization using weighted sum Artificial Bee Colony algorithm for Load Frequency Control, *International Journal of Electrical Power and Energy Systems*, 55, 657-667 (2014).
- [12] Hung M.-H., Shu L.-S., Ho S.-J., and Hwang S.-F., A Novel Intelligent Multiobjective Simulated Annealing Algorithm for Designing Robust PID Controllers, *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 38, 2, 319-330 (2008).
- [13] Tseng C.-S., and Chen B.-S., Multiobjective PID control design in uncertain robotic systems using neural network elimination scheme, *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 31, 6, 632-644 (2001).
- [14] Takahashi R.H.C., Peres P.L.D., and Ferreira P.A.V., Multiobjective H_2/H_∞ guaranteed cost PID design, *IEEE Control Systems*, 17, 5, 37-47 (2002).
- [15] Tang K.S., Man K.F., Chen G., and Kwong S., An optimal fuzzy PID controller, *IEEE Transactions on Industrial Electronics*, 48, 4, 757-765 (2001).
- [16] Liua G.P., and Daley S., Optimal-tuning PID control for industrial systems, *Control Engineering Practice*, 9, 11, 1185-1194 (2001).
- [17] Ayalaa H.V.H., and Coelho L.S., Tuning of PID controller based on a multiobjective genetic algorithm applied to a robotic manipulator, *Expert Systems with Applications*, 49, 10, 8968-8974 (2012).
- [18] Panda S., Multi-objective PID controller tuning for a FACTS-based damping stabilizer using Non-dominated Sorting Genetic Algorithm-II, *International Journal of Electrical Power and Energy Systems*, 33, 7, 12961308 (2011).
- [19] Bourmistrova, A., Storey, I., and Subic A., Multi-objective optimisation of active and semi-active suspension systems with application of evolutionary algorithm, *International Conference on Modelling and Simulation*, 1217-1223 (2005).
- [20] Xue X.D., Cheng K.W.E., and Xu C.D., Optimization of spring stiffness in automotive and rail active suspension systems, *International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles, International Transportation Electrification*, 1-6 (2016).

- [21] Aldair A.A., Abdalla T.Y., and Alsaedee E.B., Design of PID controller using ABC for full vehicle nonlinear active suspension system with passenger seat, *International Journal of Computer Applications*, 167, 18-27 (2017).
- [22] Rastegar N., and Khorram E., A combined scalarizing method for multiobjective programming problems, *European Journal of Operational Research*, 236, 1, 229237 (2014).
- [23] Haimes Y., Ladson L., and Wismer D., On a bi criterion formulation of the problems of integrated system identification and system optimization, *IEEE Transactions on System, Man, and Cybernetics*, 1, 296-297 (1971).
- [24] Benson H.P., Existence of efficient solutions for vector maximization problems, *Journal of Optimization Theory and Applications*, 26, 4, 569-580 (1978).
- [25] Benson H.P., and Morin T.L., The vector maximization problem: Proper Efficiency and Stability, *SIAM Journal of Applied Mathematics*, 1-18 (1974).
- [26] Soleimani-Damaneh M., and Zamani M., On Bensons scalarization in multiobjective optimization, *Optimization Letters*, Early Access, 1-6 (2016).
- [27] Ehrgott M., Approximation algorithms for combinatorial multicriteria optimization problems, *International Transactions in Operational Research*, 7, 531 (2000).
- [28] Deb K., *Multi-Objective Optimization Using Evolutionary Algorithms*, Publisher : Chichester, UK : Wiley (2002).
- [29] Kennedy J., and Eberhart R.C., Particle Swarm Optimization, *Proceedings of IEEE International Conference on Neural Networks*, 1942-1948 (1995).
- [30] Storn R., and Price K., Differential Evolution - a simple and efficient adaptive scheme for global optimization over continuous space, *International Computer Science Institute, Technical Report*, TR-95-012 (1995).
- [31] Yang X.-S., Firefly Algorithm, *Nature Inspired Metaheuristic Algorithms*, 79-90 (2008).
- [32] Fister I., Yang X.-S., and Brest J., A Comprehensive Review of Firefly Algorithm, *Swarm and Evolutionary Computation*, 13, 34-46 (2013).
- [33] Yang X.-S., Firefly Algorithm, *Stochastic Test Functions and Design Optimization*, *International Journal of Bio-Inspired Computation*, 2, 78-84 (2010).
- [34] Yang X.-S., *Firefly Algorithms for Multimodal Optimization*, *Lecture Notes in Computer Science*, 5792 LNCS, 169-178 (2009).
- [35] Yang X.-S., Hosseini S., and Gandomi A.H., Firefly Algorithm for Solving Non-convex Economic Dispatch Problems with Valve Loading Effect, *Applied Soft Computing Journal*, 12, 1180-1186 (2012).
- [36] Reynolds R.G., *An Adaptive Computer Model of the Evolution of Agriculture for Hunter-Gatherers in the Valley of Oaxaca Mexico*, PH.D. Thesis, Department of Computer Science, University of Michigan, (1976).
- [37] Reynolds R.G., and Peng B. *Cultural Algorithms: Modeling of How Cultures Learn to Solve Problems*, *International Conference on Tools with Artificial Intelligence*, 1-7 (2004).
- [38] Rychtyckyj, N., *Using Cultural Algorithm in Industry*, *IEEE Swarm Intelligence Symposium*, 1-8 (2003).
- [39] Chung C., and Reynolds R.G., CAEP: An Evolution-based tool for Real-Valued Function Optimization using Cultural Algorithms, *International Journal on Artificial Intelligence Tools*, 7, 239-291 (1998).
- [40] Jin X., and Reynolds R.G., *Using Knowledge-based Evolutionary Computation to Solve Nonlinear Constraint Optimization Problems: A Cultural Algorithm Approach*, *Congress on Evolutionary Computation*, 1672-1678 (1999).
- [41] Reynolds R.G., and Saleem S. *The Impact of Environmental Dynamics on Cultural Emergence*, *Oxford University Press*, 1-10 (2003).

O. Tolga Altınöz received his B.S. and M.S degrees in Department of Electrical and Electronics Engineering from Baskent University in 2003 and 2009 respectively, and Ph.D. degree from Ankara University in 2015 from Department of Electrical and Electronics Engineering. He is currently with the same department in Ankara University. His current research interests include evolutionary computation, optimization, control systems, power electronics, machine learning, and biomedical systems.

An International Journal of Optimization and Control: Theories & Applications (<http://ijocta.balikesir.edu.tr>)



This work is licensed under a Creative Commons Attribution 4.0 International License. The authors retain ownership of the copyright for their article, but they allow anyone to download, reuse, reprint, modify, distribute, and/or copy articles in IJOCTA, so long as the original authors and source are credited. To see the complete license contents, please visit <http://creativecommons.org/licenses/by/4.0/>.