Performance analysis for an improved strategy in optimal control of civil engineering structures

F Păuleț-Crăiniceanu, V Florea, C Pastia, S G Luca and O V Roșca
“Gheorghe Asachi” Technical University of Iași, 1 Mangeron Blvd., Romania

E-mail: fideliu.paulet-crainiceanu@academic.tuiasi.ro

Abstract. Civil Engineering structures are already an important part in the Smart Cities concept. In connection to this, integration of new technologies is under way in the field of Structural Engineering as in all Civil and Buildings/Bridges Engineering. One of the main goals is to protect structures from possible harm from earthquakes. In order to achieve this goal, Optimal Active Control has been proposed and used with a high degree of success. However, the application strategies are continuously improving. A strategy consisting in simplifying the Optimal Active control for full states has been proposed by the authors and relatively recently modified for an increased efficiency. That was a theoretical research taking into account the possibility to reduce the order of magnitude of matrices involved in computations. The present paper is assuming the previous theoretical result and simulates its application to the structural model of a ground floor plus ten floors building specially designed for this purpose. The reinforced concrete frame building's behaviour is analysed under seismic actions. Efficiency and the control strategy are observed in different scenarios. The structural response is revealing that the new strategy is working faster in limiting the effects of earthquake input. In addition, the reduced computational time leads to a faster control and smaller phase-shift in practical use.

1. Introduction
The actual human society is relying more and more on technological advances that must lead to an improved assurance of community safety and functionality. In the case of the Civil Engineering domain, the population is expecting to use buildings and bridges with the highest standards of strength and comfort even during and after very strong natural or anthropic actions.

For the earthquake effects mitigation, a solution is to provide flexible constructions with mechanical and computational means that prevent damages and losses while permitting the functionality to be uninterrupted. In this line, the use of passive and active methods has been intensively studied and, more or less experimentally, applied during the last decades [1, 2].

Though the cost of using structural control is still high, the last years have shown that the price of many electronics components – as sensors, communication devices, processors – and advanced mechanical components – as Tuned Mass Dampers (TMDs) or Active Mass Dampers (AMDs) – has dropped dramatically while their quality has been increased. This obvious trend is leading to keep researches on the topic [3, 4].

One of the problems in the case of active structural control applied to structures under strong seismic input is to get better methodologies leading to better performances, especially for large structures with many devices, e.g. [5, 6], where computational time is critical.

In a relatively recent work, [7], the active structural method developed in [5] and shown in detail in [6] was theoretically revised and improved. The improvement is based on considering the particular case of proposed control matrices and the calculus of the gain matrix that contain some zero submatrices. That was leading to the conclusion of solving two Riccati matrix equations (one for symmetrical and one for non-symmetrical solutions) instead of a double-dimensional Riccati symmetrical matrix equation.

This scale down of the main problem is now debated in terms of a numerical study taking into focus a ground floor plus ten stories flexible building, designed for this study.
2. Theoretical background

To implement active structural control in Civil Engineering using Active Mass Dampers (AMDs), the full state Linear Quadratic Regulator method was used [5] with some simplifications [6].

In this method, the central point from computational point of view is to solve the $2n$-dimensional Riccati matrix equation

$$PA - PBL_i R^{-1}L_i^t B' P + A' P + Q = 0$$

where $P$ is the unknown $2n \times 2n$ matrix and where $n$ is the number of dynamic degrees of freedom involved by the structure in the classical Structural Dynamics description.

In equation (1), the matrices $A$ and $B$ are

$$A = \begin{bmatrix} 0 & I \\ -M_s^{-1}K_s & -M_s^{-1}C_s \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ M_s^{-1} \end{bmatrix}$$

where $M_s$ is the $n \times n$ structural mass matrix, $C_s$ is the $n \times n$ structural damping matrix and $K_s$ is the $n \times n$ structural stiffness matrix.

In addition, in equation (1) the $2n \times 2n$ weighting matrix $Q$ is imposed as

$$Q = \begin{bmatrix} K_s & 0 \\ 0 & M_s \end{bmatrix}$$

and the $m \times m$ weighting matrix $R$ as

$$R = \text{diag} \{ r_1, \ldots, r_m \}$$

where $m$ is the number of active devices, AMDs in this application. For a simpler implementation, the matrix $R$ is iteratively obtained through trials on only one scalar $r$ on its diagonal

$$R = rI$$

and $I$ is the identity matrix having $m \times m$ dimension.

To finish the description of equation (1), it should be noted that $L_i$ is an $n \times m$ matrix showing the distribution of active forces on structural degrees of freedom.

In this paper, the full states control low is implemented as

$$u = L_i K x$$

where $u$ is the $n \times 1$ control forces vector, time dependent, and $K$ is the $m \times 2n$ control (gain) matrix, with $K = R^{-1} L_i B' P$. 

As shown in [7], solving the $2n$-dimensional Riccati equation (1) for a symmetric solution $P$ might be replaced with solving two $n$-dimensional Riccati equations:

$$P_{12} A_{11} - P_{12} B R^{-1} B' P_{21} + A' P_{21} + Q_1 = 0$$

and

$$P_{21} + P_{22} A_{22} - P_{22} B R^{-1} B' P_{22} + P_{12} + A' P_{22} + Q_2 = 0$$

where notations are from decomposition of the $2n \times 2n$ matrix $P$, the symmetric solution of equation (1), into four $n \times n$ submatrices.
\[ P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \]  

(9)

also from decomposing \(2n \times 2n\) matrix \(Q\) in halves

\[ Q = \begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix} \]  

(10)

and, similarly, decomposing

\[ A = \begin{bmatrix} 0 & I \\ A_{21} & A_{22} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ M_k^T L_k \end{bmatrix} = \begin{bmatrix} 0 \\ B_2 \end{bmatrix} \]  

(11)

Knowing that \(P_{12} = P_{21}'\), the first equation of the two equations above, (7) and (8), is solved for \(P_{12}\) and the result is used in equation (8). While equation (7) has a non-symmetric solution, equation (8)’s solution, \(P_{22}\), is symmetric.

In this paper, the theoretical approach showing that the gain matrix, \(K\), can be obtained using only \(n\)-dimensional equations and is validated through a numerical case study in the next section. However, it must be noted that aspects of the efficiency in computation are rising new challenges for future studies.

3. Numerical simulation and performance analysis

In order to numerically validate the above theoretical strategy, the goal of simulating the active control is eleven stories building designed for this case. As computational tool, a personal computer running Matlab [8] was employed. Results are shown the validity and the interesting aspects concerning the proposed strategy.

The seismic action used for this application is the north-south component of the acceleration recorded during the 1977 Vrancea, Romania, earthquake, shown in Figure 1.

![Figure 1. The seismic input used by numerical simulation.](image)

3.1. The structure involved in simulation

For achieving the goals of this study (numerical validation and the analysis for computational performance of the strategy proposed in [7]), the structural matrices from a finite element method
(FEM) model shown in Figure 2 were imposed. This structure has a ground floor plus ten floors, the last one being a smaller, technical one.

The structure is composed from 206 nodes, first 20 nodes being the connection to foundations. Therefore, the number of equations generated for this model is 1116. A number of 472 beams are also part of the model. The number of lumped masses is 372, equal to the number of dynamic degrees of freedom that constitute also the dimension of the classic structural dynamic’s matrix problem. Adding the dynamics of the eight TMDs / AMDs will increase the size of the problem to 380.

![Figure 2. Structural model used in numerical simulation, during vibration.](image)

In order to serve the present work, the structure is a flexible one with the first period of vibration equal to \(T=1.301\) s. The proposed control system is strengthening the structure.

3.2. TMDs and AMDs characteristics

The numerical simulation employed a number of eight identically TMDs. They are located in a number of four at levels 7 and 9 over the ground floor. Each of these TMDs are characterised by the 4-t mass, 100 kN/m stiffness and 10% damping, resulting in a tuned period of vibration 1.257 s, closer to 1.301 s, [8], the first structural natural period of vibration. Masses of the TMDs total 0.38% of the building’s mass. The action of these TMDs is along the X axis direction of the building model. The case of using TMDs is denoted as the “Passive control” case of computer runs in this paper.

Later, the TMDs are transformed in AMDs, involving actuators that can move the masses conforming to the control law calculated by equation (1) and referred as “Control 1”. Then, the computations were under the control law implying the solution proposed in [7], and shown by equations (7) and (8), situation labelled as “Control 2” in this study. The TMDs and corresponding AMDs are numbered from 1 to 8, first four being located at the level 7 and the others at the level 9 above the ground floor.

3.3. Programming the problem

In order to simulate all the planned situations (“No control”, “Passive control”, “Control 1” and “Control 2”), Matlab computer software was run [9].

The “No control” program was run in order to study the structural characteristics of the structure and to obtain the corresponding frequency response and seismic time response to Vrancea 1977 NS component earthquake. In addition, comparisons of the two types of responses have been done. Comparisons for each level frequency response were also performed. Responses were calculated in terms of displacements, velocities and accelerations.
A second computer program was targeting the “Passive control” scenario, when the 8 tuned masses were placed on the structure, acting on x direction. The very same time of responses as in the case of “No control” scenario had been issued. Comparisons “No control” versus “Passive control” were also done.

The third Matlab program (referring to the “Control 1” scenario) is concerned with the Linear Quadratic Regulator involving the solution of the equation described in equation (1), i.e. using 2n-dimensional matrices. In this case, two ways to calculate the solution of the Riccati equation (1) had been employed:

1. using the \texttt{lqr} Matlab function (with calculus of the gain matrix) that is based on a general solution method and implying a longer time in computation
2. using the \texttt{Riccati} function for a symmetric 2n-dimensional solution obtained from \textit{MATLAB Central File Exchange} [10], shown in the Appendix.

For the “Control 2” scenario, a Matlab program was created for solving the two equations, (7) and (8). These equations were solved through:

1. using \texttt{Riccati} function for symmetrical solution (n-dimensional) [10], for equation (7)
2. using \texttt{Solve_BQMEs} function, shown in Appendix, for a non-symmetrical solution (n-dimensional) of equation (8). This function is adapted from the original one from \textit{MATLAB Central File Exchange}, [11].

3.4. Computational means

The computer used in this study was a personal computer with the next main computational characteristics: Intel I5-7400 processor clocked at 3 GHz with 4 cores; 32 GB RAM; 1TB SSD; Radeon RX580 video card with 8GB GDDR5 RAM.

For obtaining the results, the runs were done under the “same” (as possible) conditions, i.e. without the interference of other programs or internet connection.

3.5. Results and comments

From the runs, a series of selected results show the quality of the control strategy (with better responses compared to passive and non-controlled cases) together with the comparisons between the “Control 1” and “Control 2” scenarios.

The responses in the two control situations are practically identical in all cases (time responses, responses to harmonic loads and maximum responses).

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{structure_displacement.png}
\caption{Level 9’s displacement on the four scenarios.}
\end{figure}
Figures 3 and 4 prove the above statement. There, the seismic time response under Vrancea NS 1977 earthquake in terms of displacement and acceleration for the level 9 of the building is shown in the four scenarios taken into views of this study. In addition, the cases without devices (“No control”) and with TMDs (“Passive case”) are presented.

In addition, it can be seen that the passive case is not too effective. This is because the number of devices is not enough to lead to better response. Adding more devices would have increased the total additional mass, creating an unfavourable situation. It should be mentioned that the role of this study is to focus on active control. The passive case is used mainly as a reference for comparisons with no-control case. Figures 3 and 4 are showing the almost identical responses in the two control cases, proving that the two computational versions presented in section 2 have the same performance level.

**Figure 4.** Level 9’s acceleration on the four scenarios.

**Figure 5.** Maximum acceleration on level in the four cases.

Figures 5 and 6 prove the same. The performance of the control strategies is shown in terms of maximum accelerations during the earthquake action at level 9 above ground floor. The first of the
two figures refer to absolute floor displacement, while in the second figure the storey drift is the element of comparison.

![Comparison of maximum storey drift on x](image)

**Figure 6.** Maximum storey drift on level in each of the four scenarios.

In Figures 7 and 8, the frequency response is in use for proving the correctness of the theoretical proposals in Section 2. Figure 7 shows that the control strategies are very close in performance and much better than the other two results from the non-control and passive cases. If Figure 7 is making comparisons in terms of accelerations, Figure 8 is employing the same kind of comparisons in terms of velocity of the ninth floor of the structure used in this study.

![Acceleration response spectra](image) ![Velocity response spectra on x](image)

**Figure 7.** Level 9’s acceleration response to harmonic loads in the four scenarios. **Figure 8.** Comparison between the floors’ response to harmonic loads in velocity terms.

Figure 9 adds insights of the control strategies requirements in terms of actuating need of forces (showing again the almost no differences between the “Control 1” and “Control 2” cases). This figure is stressing once more on the validity of the proposed strategy because the control 1 and 2 versions are showing very close effectiveness for the upper floor responses.
Figure 9. A typical actuator requirement in terms of forces.

Figure 10 is a comparison in terms of displacements of TMS versus AMD case (in order to observe the need of large displacements required by active control compared to the passive control). This figure helps understanding the mechanism of control imposed by the control law. Also, as in the other figures, the almost no difference between the two control strategies is adding more confirmation to the methodology.

Figure 10. Comparison between control 1 and 2 with AMDs and TMDs movement.
In conclusion, the numerical validation of the “Control 2” scenario is strongly established. Regarding the time efficiency of the “Control 2” scenario over the “Control 1” scenario, the discussion must be based on numbers presented in Table 1.

Table 1. Running times for the control strategies, in seconds.

| Run no. | Control 1\(^a\) | Control 2\(^b\) | Performance\(^c\) |
|---------|-----------------|-----------------|-------------------|
|         | lqr\(^d\)      | Riccati-s\(^e\) | Bome-n\(^f\)      |
| 1       | 57.5205         | 1.87137         | 1.55026           |
| 2       | 57.5070         | 1.88718         | 1.51286           |
| 3       | 57.9074         | 1.88794         | 1.48044           |
| 4       | 57.4805         | 1.87917         | 1.47524           |
| 5       | 57.5284         | 1.88980         | 1.46890           |
| 6       | 57.4184         | 1.88706         | 1.47544           |
| 7       | 57.9965         | 1.89199         | 1.47898           |
| 8       | 57.4732         | 1.86806         | 1.47985           |
| 9       | 57.3530         | 1.87716         | 1.48543           |
| 10      | 58.4764         | 1.92876         | 1.46947           |
| Mean    | 57.6662         | 1.88685         | 1.48770           |

\(^a-k\) See explanations in text, paragraph 3.5.

Table 1 shows groups of 10 simulations time needed for running scenarios for validating the control proposed strategy. Therefore, the next items had been taken into account:

a) First control strategy, as described in [5, 6]. The involved matrices are 2n-dimensional (i.e. 760 x760 elements for the solution matrix).

b) Second control strategy, as described in [7]. The involved matrices are n-dimensional (380x380 elements for solution matrices).

c) Performance evaluated as in differences in values in “Control 2” minus values in “Control 1”.

d) Run time, in seconds, using lqr Matlab function.

e) Run time, in seconds, using Riccati function for symmetrical solution (2n-dimensional) [10].

f) Run time, in seconds, using Solve_BQME function for non-symmetrical solution (n-dimensional) [11].

g) Run time, in seconds, using Riccati function for symmetrical solution (n-dimensional) [10].

h) Run time, in seconds, for calculus of the gain matrix in the case of “Control 2” strategy.

i) Total run time for “Control 2” (column f + column g + column h).

j) Performance evaluation in absolute running time, in seconds (column i – column e).

k) Performance evaluation in relative running time, in percent, i.e. (column i – column e)/column i*100.

From Table 1, it can be withdrawn that the time for running the lqr function (for the 760x760 size solution) is very long, more than 57 seconds on average (column d). More efficient is to use in this case the Riccati function [10] for the same size of symmetric solution matrix. An average of 1.88685s was measured (column e).

From the same Table 1, it can be observed that dividing the problem in two smaller Riccati-like problems leads to a small efficiency in time (an average of only 3.7%, column k) compared to the initial problem, column e. However, the use of Riccati function for the half problem (380x380 size symmetric solution, column g) is showing an average of 0.32738s (column g) compared to 1.88685s (column e) for the 760x760 size solution. In other words, the efficiency in time is around 1/6.
A conclusion from above aspects in Table 1 is that for non-symmetric solution, column $f$, the function $\text{Solve\_BQMEs}$ (shown in the Appendix) should get some further refinement or should be replaced by a more efficient function. This is the point to focus on in the near researches.

4. Conclusions

In a previous work, [7], a proposal for improving the algorithm of optimal active control in Civil Engineering structures has been shown. The proposed approach is asking for solutions of two $n$-dimensional Riccati matrix equations, one with a symmetrical solution and another with a non-symmetrical solution. This is done in comparison to classical $2n$-dimensional problem with symmetrical solution.

The numerical simulation of this proposed approach was applied to a ground level plus ten stories building. The numerical results, using the classical method and the proposed one, presented practical no differences, this way validating the method.

This validation is proved by the presented figures. Seismic time responses of the structure, together with the spectral responses or maximum story displacements and storey drifts are proving that the two approaches have almost the same performance.

From the computational point of view, a very good improvement in time computation is observed in comparison to $\text{lqr}$ Matlab function applied to the original $2n$-dimensional problem. However, in the case of using other functions (as $\text{Riccati.m}$ calling Schur approach for symmetrical solutions together with $\text{Solve\_BQME.m}$ - an iterative adapted function for non-symmetrical solutions, shown in the Appendix) the performances are closer but still better (lower time) with an average of around -3.7% in favour of the proposed methodology.

As a consequence, the method being numerical validated, the future researches will focus on obtaining faster computational means for matrical Riccati equations applied to non-symmetrical solution cases. In addition, based on previous work, e.g. [12], more practical features might be added to the procedure.

Acknowledgments

Authors wish to acknowledge and be grateful for the kind support from The MathWorks, Inc. that provided full Matlab and Simulink software.

5. References

[1] Ghaedi K Ibrahim Z Adeli H and Javanmardi A 2017 Invited Review: Recent developments in vibration control of building and bridge structures Journal of Vibroengineering 19 5 p 3564-3580
[2] Li H and Huo L 2010 Advances in Structural Control in Civil Engineering in China Mathematical Problems in Engineering 2010 936081 pp 1-23
[3] Preumont A Seto K 2008 Active Control of Structures (Chichester: JohnWiley & Sons)
[4] Xu Y L and He J 2017 Smart civil structures (Boca Raton: Taylor & Francis Group)
[5] Paulet-Crainiceanu F 1997 Active Control Approach for Long Span Bridge Responses to Strong Earthquakes Ph.D. Thesis (Yokohama: Yokohama National University)
[6] Paulet-Crainiceanu F 1999 Seismic Response Control of Long Cable-stayed Bridges Proc. Of the 2nd World Conf. on Structural Control (Kyoto) (Chichester: John Willey & Sons) vol 2 p 959-964
[7] Paulet-Crainiceanu F, Luca S-G, Pastia C and Florea V 2019 Improved strategy for optimal control of Civil Engineering structures 15th Int. Conf. on Computational Civil Engineering (Iasi) IOP Conference Series: Materials Science and Engineering Vol 586 pp 1-6 (IOP Publishing Ltd)
[8] Pastia C Luca S G Paulet-Crainiceanu F and Florea V 2016 Passive, Active and Semi-Active Vibration Control of a Classical Mass-Spring-Dashpot System 16th Int
Multidisciplinary Scientific GeoConference SGEM2016 (Sofia) (Sofia: Stef92 Technology Ltd.) book1 vol. 3 p 575

[9] The MathWorks Inc Matlab R2021a, www.mathworks.com
[10] Emmett 2021 Algebraic Riccati Equation Solver MATLAB Central File Exchange (https://www.mathworks.com/matlabcentral/fileexchange/36263-algebraic-riccati-equation-solver) Retrieved March 20, 2021
[11] Sadek L 2021 Nonsymmetric differential Riccati matrix equations MATLAB Central File Exchange (https://www.mathworks.com/matlabcentral/fileexchange/72895-nonsymmetric-differential-riccati-matrix-equations) Retrieved March 20, 2021
[12] Bakule L Paulet-Crainiceanu F Rodellar J and Rossell J M 2005 Overlapping Reliable Control for a Cable-Stayed Bridge Benchmark IEEE Trans. C.S.T. 13 4 663–9

Appendix. Some functions used in simulations
The modified Solve_BQME.m, [11], used under the name Solve_BQMe.s.m used together with the also modified EvalBiQuad.m function:

```matlab
function X=Solve_BQMe.s.m(A,B,C,D)
% This Function is adapted to solve the equation
% of the form A'X'+XB+XCX'+D = 0 for non-symmetric X
[p1,p2]=size(A);
[p3,p4]=size(B);
X=zeros(p2,p3);
itors=0;
Omega=EvalBiQuad(A,B,C,D,X);
while (itors<100*sum(size(X))) && (norm(Omega)>1e-9)
Delta = sylvester(A'+X*C,B+C*X',-Omega);
if norm(Delta) > 1e10
error('Cannot Find Real Valued Solution')
end
itors=itors+1;
X=X+Delta;
Omega=EvalBiQuad(A,B,C,D,X);
end

function Omega=EvalBiQuad(A,B,C,D,X)
Omega=A'*X'+X*B+X*C*X'+D;
end
```

The next function [10], Riccati.m, solves the classical matrical Riccati equation with symmetrical solution using the Schur method:

```matlab
function X=Riccati(A,G,Q)
%RICCATI Solves an algebraic Riccati equation
% X = Riccati(A,G,Q) solves the algebraic Riccati equation of the form:
% A'*X + X*A' - X*G*X + Q = 0, where X is symmetric.

n=size(A,1);
Z=[A -G
   -Q -A'];
[U1,S1]=schur(Z);
[U,S]=ordschur(U1,S1,'lhp');
X=U(n+1:end,1:n)*U(1:n,1:n)^-1;
end
```