ASSET MANAGEMENT OPTIMIZATION MODELS: MODEL SIZE REDUCTION IN THE CONTEXT OF PAVEMENT MANAGEMENT SYSTEM

Abhishek Bhargava *
Consultant, Ph.D., AgileAssets Inc. (PKA Texas Research and Development Inc.), USA
* 3200 Atlantic Avenue, Suite 103, Raleigh, NC 27606, USA abhargava@agileassets.com

Alexander Galenko
Operations Research Analyst, Ph.D., AgileAssets Inc. (PKA Texas Research and Development Inc.), USA

Tonya Scheinberg
Senior Vice President, Ph.D., AgileAssets Inc. (PKA Texas Research and Development Inc.), USA

ABSTRACT: In this paper the authors present a number of ways that can be used to substantially reduce the General Asset Management Problem size. Four size-reducing optimization techniques are presented. These techniques then are evaluated and compared with each other and with the complete full-size General Asset Management Problem in terms of the gap between the optimal solution and the solutions of reduced-size problems. The impact of the various reduction techniques on the solution is analyzed. In this new era of limited transportation funding the results presented in this paper will encourage agencies and researchers world-wide to reexamine and reevaluate the effectiveness of the recommended optimal work plans by their management systems.

KEY WORDS: Pavements, optimization, maintenance, management, asset, model

1. INTRODUCTION

The lack of sufficient resources to meet transportation needs brought on by the Great Recession and aging infrastructure has sparked a policy shift in Transportation Infrastructure Asset Management. Over the years, there has been an increased focus on development of performance based (for example, signed into a public law the Moving Ahead for Progress in the 21st Century Act in the USA [21]) optimal investment strategies. Optimization based Asset Management models are increasingly being sought by transportation agencies around the world, to come up with optimal project selection algorithms, and, consequently investment strategies for each infrastructure asset.

Just finding an optimal maintenance plan for one asset over let’s say 20 years can be a computationally expensive exercise. Considering simultaneously several thousand such assets and controlling their overall performance trend to make most sophisticated and “smart” models technologically infeasible pushing practitioners to use less sophisticated models that are proven to produce inferior maintenance plans. In the end this results in inefficient long term budgeting.

Typically Asset Management problems are formulated as a Mixed-Integer Problem (MIP). An instance of a multi-year problem that uses MIP formulation and finds an optimal maintenance plan, for example, for a road system can easily have over several billion integer variables. Even the best solvers struggle to solve problems of that size. In this situation the use of meta-heuristics, such as genetic algorithms, seems to be somewhat justified even though they do not give any information about the quality of their solution. On the other hand, it is possible in some cases to reduce the initial problem size in such a way that the use of optimal solvers becomes feasible. Surprisingly, these techniques are not widely used.

Indeed, the majority of published literature is focused on modeling part of the General Asset Management Problem (GAMP) or on optimization algorithms that can be used to solve it. On the modeling side there are three major classes of models: Prioritization, Single Asset Life-cycle composition and Integer Programming. Each of these models can be applied to both deterministic and stochastic problems.
Prioritization models are the simplest. The main goal of these projects is to derive a synthetic rank for each project given its various state variables and then select only those that have the highest rank and can be executed for a given budget. For example, see works by Moazami et al. [13] and Gabriel et al. [7]. Moazami et al. [13] developed a prioritization model based on fuzzy logic. Gabriel et al. [7] developed a procedure to rank projects with competing objectives under uncertainty.

Single item life-cycle composition is focused on finding optimal maintenance actions for a single asset in order to guarantee a given performance level over its life-time while minimizing the costs/risks. Here costs can reflect costs of the agency that maintains the asset as well as the costs of its users associated with possible asset failures. More recently, Gu et al. [8] presented an analytical approach for planning of pavement maintenance and resurfacing actions that minimizes pavement lifecycle costs. Okasha and Frangopol [16] developed a detailed computational framework to manage bridge lifecycles that included the assessment of life-cycle performance, analysis of system and component performance interaction. Li and Madanu [11] presented an uncertainty-based methodology for highway project level life-cycle benefit/cost analysis including an excellent literature review of lifecycle composition.

The Integer Programming models are the hardest to solve. They optimize system-wide performance over a given time horizon. They include single item life cycle composition for each asset as well as performance constraints for an entire system. See Dahl and Minken [3], Lomptey et al. [10], Wang et al. [22], Scheinberg and Anastasopoulos [20] for interesting model formulations. De La Garza et al. [4] provide a good overview of prior research in network-level optimization.

As for the optimization algorithms it is common to use the Sorting/Knapsack Solver for the Prioritization models. Single item life-cycle models are typically solved using a Dynamic Programming Algorithm, a Mixed-Integer Programming (MIP) solver or a Genetic Algorithm. Integer Programming problem’s solutions rely on a MIP solver or Genetic Algorithms.

In this paper the authors do not present a novel optimization algorithm but instead present a number of ways that can be used to substantially reduce the GAMP size. We believe these techniques if used by practitioners will result in better maintenance plans and hence more efficient resource allocation. For example, the techniques described here allowed AgileAssets to switch to a more complicated multi-year analysis model which in some cases resulted in savings of about 15% over using a simpler single year model to maintain the given road network at a particular desired performance level.

The paper is organized as follows. In the following section, the GAMP is presented. Next, several approaches to reduce GAMP size are discussed. Four size reducing techniques are presented and evaluated by comparing their solutions with the truly optimal solution that is obtained by solving GAMP as a whole.

2. GENERAL ASSET MANAGEMENT PROBLEM (GAMP)

Several formulations of the GAMP are possible. In our formulation the aim is to seek strategies – $T$-long sequence of interventions for every asset so that some asset attribute is minimized / maximized, subject to other attributes being restricted by linear constraints. The solution when using such a formulation is a maintenance plan – set of particular interventions to be applied to particular assets at specific times. The slice of this work plan for one asset is a winning strategy for that asset.

The following is the MIP GAMP formulation based on maintenance strategies.

Sets:
- $I$ - Set of assets (for example, road sections).
- $J_i$ - Set of available strategies for asset $i$.
- $T$ - Optimization time horizon.
- $A$ - Set of asset attributes under consideration.
A_{i,j,t} - Value of attribute $A$ for asset $i$ under strategy $j$ for years $t = 1, ..., T$.

$K$ - a set of rules for building optimization constraints. It includes:

$C^k$ - limit value of constraint ‘$k$’ (often called Right Hand Side or RHS), and

$R_{i,j,t}$ - area of application or feasible region which is generally defined by the constraints that are placed on the optimization problem (For example, budget / performance constraints by year, district, functional class, or other similar attributes of an asset).

The problem has only binary decision variables $x_{i,j}$:

$$x_{i,j} = \begin{cases} 1, & \text{if strategy } j \text{ is selected for asset } i \\ 0, & \text{otherwise} \end{cases}$$

The objective function (and each essential Constraint) is associated with some asset attribute so that coefficients are obtained from asset attribute values. For example, if the objective is to maximize the weighted (by attribute Weight) average of attribute PCI then $PCI_{i,j}$, Weight will give us objective coefficients $Obj_{i,j}^j$ for strategy $j$ of asset $i$ in year $t$.

As examples of essential constraints we can consider budget constraints:

$$\sum_{i,j \in R} Cost_{i,j} x_{i,j} \leq B,$$

which restricts the total cost of interventions in $R$. $R$ is the area of constraint application. It can be one year or/and geographical ownership of assets or/and special group of interventions.

$$\sum_{i, j \in R, PCI_{i,j} \geq \text{Thr}_{i}} LaneMiles_{i,j} x_{i,j} \leq PCI_{R} / \sum_{i \in R} LaneMiles_{i},$$

is another example. It restricts percentage of “fair or better” lane miles of roads in area $R$, and so on.

Given all the above The Deterministic GAMP with $K$ constraints formulated as the following:

$$\min_x \max_x \left( \sum_i \sum_{j \in J} \sum_{t \in T} Obj_{i,j}^j x_{i,j} \right)$$

Subject to

$$\sum_{i,j \in R_k} Const_{i,j} x_{i,j} \leq (\geq) C^k, k \in K$$

$$\sum_{j \in J_i} x_{i,j} = 1, i \in I$$

$$x_{i,j} \text{ binary}, i \in I, j \in J_i,$$

where the last $I$ constraints are added to guarantee that one and only one strategy will be chosen for every asset. So, the model is minimizing (maximizing) the sum of objective coefficients in $Obj_{i,j}^j$ while satisfying $K$ constraints and choosing exactly one strategy for each asset.

Here it’s worth mentioning that if one knows the current value of attribute $a_{i,0}$ of asset $i$ and a particular strategy that will be applied to that asset, then one can compute (either in deterministic or stochastic terms) future values (distributions) of attribute $a - a_{i,t}, t \in T$.

An example of attribute values calculated for a strategy is presented in Table 1. The 5-year road section strategy in this case is: Maintenance in year 1, Major repair in year 3. To compute attribute values in accordance with a strategy we use improvement formulae with each non trivial intervention. Improvement formulae allow calculation of attribute values after intervention by attribute values before the intervention. With or without intervention, performance (deterioration) models and other rules are used to “age” asset attributes one year.
In this paper we focus on the Deterministic Asset Management Model, however, the stochastic case can be formulated similarly and proposed techniques in this paper can be applied to the stochastic case.

We think it is worthwhile to point out that our formulation is a multi-year problem even though decision variables \( x \) do not have an index over \( t \). The information on when to do which intervention is implicitly hidden within strategies and the number of strategies \( |J| \) for each asset depends on the length of time horizon \( T \). The whole size of the solution space equals to the total number of all strategies for all assets – sum of \( |J| \)'s over all \( i \)'s. For the Pavement Management Case, set \( I \) contains road sections. For example, let’s say that the size of \( I \) is 7,000. The optimization time horizon \( T \) is usually 10 to 20 years. Further, let’s say that there are \( Intr \) (including “do-nothing”) interventions that can be performed on a pavement section. The size of \( Intr \) is about 30. Let’s assume that any possible intervention can be performed on any pavement section, in any year. With this assumption \( |J| = |Intr|^{T} \) for each asset. Thus, the number of binary variables in the problem is equal to \( |I| |J| = |I||Intr|^T \). In numbers it is \( 7000 \times 30^{10} \) or more than \( 10^{18} \) - obviously an unrealistic size. The number of constraints is \( |I| \) plus whatever we have in \( K \) which is negligible in comparison with \( |I| \), so it is not more than several thousand. For example, if we consider \( |I| = 103; T = 5; |Intr| = 5 \), the number of variables in the full problem is \( 103 \times 5^5 = 321,875 \). As one can see, a typical Pavement Management instance is indeed very large and hence very hard to solve. The growth is especially rapid in \( |Intr| \) and \( |T| \). In the next sections we discuss what can be done to reduce the size of this general model and investigate how proposed heuristics will deviate from the optimal solution of GAMP.

### 3. PROBLEM REDUCTION APPROACHES

The main conclusion of the previous section is the virtual impossibility of solving the GAMP in practical cases. In this section, two approaches are discussed that are generally used to reduce problem size and still provide reasonably good decisions. The first approach is to use heuristic algorithms to find an approximate solution and the second is to impose additional assumptions on the problem so that the size of the problem is drastically reduced.

#### 3.1 Heuristic Algorithms

The most popular heuristics are: (a) use of genetic algorithms, and, (b) splitting the GAMP into \( T \) smaller problems. This paper does not discuss the genetic algorithms approach because it is still in the research stage. The second heuristic solves the GAMP for each year (by breaking it into \( T \) years) and uses the solution of the previous year as an input to the next year. This approach is discussed in more details later in this paper and is referred to as the Single-Year technique.

Neither of these two heuristic approaches provides the “true” optimal solution (that is found by solving the GAMP). In fact, the distance from the optimal solution cannot be easily identified, and to our knowledge, neither of these two heuristics has been substantially researched from this perspective before (especially in the

---

**Table 1. Example of 5-year strategy**

<table>
<thead>
<tr>
<th>Intervention</th>
<th>Maintenance</th>
<th>Plan Year 2</th>
<th>Major Repair</th>
<th>Plan Year 3</th>
<th>Plan Year 4</th>
<th>Plan Year 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pavement Age</td>
<td>22</td>
<td>23</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Remaining Service Life</td>
<td>5.7665</td>
<td>4.7665</td>
<td>15.1378</td>
<td>14.1378</td>
<td>13.1378</td>
<td></td>
</tr>
<tr>
<td>Intervention Cost</td>
<td>557793</td>
<td>0</td>
<td>3904553</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Pavement Condition Index</td>
<td>75.487</td>
<td>69.938</td>
<td>100</td>
<td>99.16</td>
<td>97.585</td>
<td></td>
</tr>
<tr>
<td>Years Since Last Treatment</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
transportation asset management context). In spite of that, the Single-Year technique is being used by most commercial transportation management systems.

3.2 Additional Model Assumptions
In this paper, two types of ‘Additional Assumptions’ are considered to reduce the size of the GAMP. The first is the assumption that ‘an asset can be treated not more than once during the analysis period $T$ (Single-Action assumption)’, and the second one is that ‘some “decision mechanism” (Tree-Action assumption) can be used to limit the number of potential interventions on a given asset in a given year’.

It is important to note that while assumptions, such as the above, can reduce the problem size substantially using them in practice often comes at the cost of a departure from an optimal solution. Therefore, it is important to have good reasoning and a clear understanding of how valid the assumption is, and what the potential impact on the optimal solution it might have (discussed later in this paper).

We believe the most promising way to reduce the number of strategies under consideration is to use a decision mechanism, i.e. an implementation of engineering solutions that allows choosing one or several possible interventions depending on values of asset attributes. It is like a manifestation of common sense - one would not consider the “Reconstruction” intervention for a perfectly good asset (and definitely not every year). A decision mechanism can be implemented with decision trees, decision matrices, neural network or any “crystal ball” that takes asset attribute values and recommends interventions. One could think of a decision mechanism as an expert who suggests what should be done given the current asset state. See Haas et al. [9], Chapter 18 for an overview of decision processes and expert systems approaches to identify feasible alternatives in the context of asset management.

4. FORMULATING REDUCED-SIZE TECHNIQUES
The problem reduction approaches discussed in the previous section were used to develop the following four reduced-size techniques. The objective was to compare and evaluate the optimal solutions of these techniques with the GAMP, and thus demonstrate how such reduced-size problems should be analyzed for performance and effectiveness.

<table>
<thead>
<tr>
<th>Heuristic</th>
<th>Assumption</th>
<th>Reduction Technique</th>
<th>Problem Size (# of Variables)</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>None</td>
<td>GAMP</td>
<td>$</td>
</tr>
<tr>
<td>None</td>
<td>Single Action</td>
<td>Multi-Year Single Action</td>
<td>$</td>
</tr>
<tr>
<td>None</td>
<td>Tree Action</td>
<td>Multi-Year Tree Action</td>
<td>$</td>
</tr>
<tr>
<td>Single-Year</td>
<td>None</td>
<td>Single-Year Multi-Action</td>
<td>$</td>
</tr>
<tr>
<td>Single-Year</td>
<td>Tree-Action</td>
<td>Single-Year Tree-Action</td>
<td>$</td>
</tr>
</tbody>
</table>

**Multi-Year Single Action**
This technique is just one simplified form of the GAMP which is derived after the following assumption is imposed: “At most one non trivial intervention per asset over optimization time horizon $T$ is allowed”. That is, every non-trivial intervention (all but ‘do nothing’) is possible every year but the asset is treated only once, all other years are filled with ‘do nothing’ intervention. With this technique the set of possible asset strategies is reduced from $|Intr|^{|T|}$ to $(|Intr|-1)|T| + 1$ (any of $|Intr|$ interventions can be used for any asset in any one year and the “do nothing” strategy is always under consideration) and optimization problem has the size $J = |I|\cdot|T|\cdot(|Intr|-1) + 1$ - more manageable number of variables than in the general problem. For example, when $I=103; T=5; Intr=5$, the problem size is $103*(5-1)*5 + 1 = 2163$, see Table 4.
Single-Year Multi-Action

The Single-Year Multi-Action technique is a simplified / reduced form of the GAMP. Essentially it is equivalent to the GAMP with \(| T | = I \). The set of interventions is still \(| Intr |\), and any of the interventions can be applied to any of the assets in each year. However, to obtain a multi-year work plan, the optimization problem is solved iteratively by taking the current year solution (and its consequences) as an input for the next year problem. As a result, with this technique one has to solve \( T \) problems, each of the size \( J = | I | * | Intr | \) - a manageable number of variables. For example, in an instance where \(| I | = 103, | Intr | = 5 \) and \( T = 5 \), the size of each of the \( T \) Single-Year Multi-Action problems is given by: \( J = 103 * 5 = 515 \) (see Table 4).

The Single-Year Multi-Action problem with \( K \) has the form:

\[
\begin{align*}
\min (\max) & \sum_{i \in I} \sum_{j \in J} Obj_{i,j} x_{i,j} \\
\text{Subject to} & \sum_{i \in I \cap j \in J} Constr_{i,j} x_{i,j} \leq (\geq) C^k, k \in K \\
& \sum_{j \in J} x_{i,j} = 1, i \in I \\
& x_{i,j} \text{ is binary}, i \in I, j \in J
\end{align*}
\]

where \( x_{i,j} = 1 \) means that intervention \( j \) is applied to asset \( i \). Additional constraints assure that one and only one intervention will be chosen for any given asset. (“do nothing” is always one of the possible interventions).

There are advantages and drawbacks to this technique. The drawback is obvious – since the optimization problem is not aware of a time component and the system dynamic over time, the final \( T \)-year plan might not be the optimal solution of the general problem and could lead to more frequent cheap interventions and hence overspending. The main advantage is simplicity. There is no need to pre-calculate attribute profiles, since every year attributes are calculated based on previous year solutions. This technique is widely used by asset management professionals.

Single-Year Tree-Action

The ‘decision mechanism’ existence assumption can be coupled with the heuristic approaches to further reduce the size of the optimization problems. One of such hybrid reduction techniques we will call the Single-Year Tree-Action. Under this technique the GAMP is broken down into \( T \) single-year problems, and a decision tree is used for each period to recommend only a subset of available interventions for every asset. The purpose of the decision mechanism is to limit the number of alternative interventions that can be applied on an asset. Using such a mechanism allows reducing the model’s size further as the number of possible interventions for each asset in a single-year problem is reduced to recommended interventions and “do nothing”.

While some decision trees reduce the number of alternative actions to one, there are others that can reduce the number of actions to three or six or any other reasonable number that is less than the size of \(| Intr |\). The size of the Single-Year Tree-Action models therefore depends on the number of interventions that are recommended by the decision mechanism (tree) for each asset. In this paper, we assume that the decision tree used in the Tree-Action models recommends one appropriate intervention for each asset based on a logical decision mechanism. Under this assumption a model derived using Single-Year Tree-Action technique will have only 103 variables (see Table 4).

Multi-Year Tree-Action

Another hybrid technique we propose is Multi-Year Tree-Action. It produces a multi-year problem which utilizes multi-year strategies. The multi-year strategies for each asset are built using decision trees that recommend a ‘single’ intervention (one expert opinion). Thus we can put the question of “what to do?” out of the way and leave only a question of “when to do?” an intervention. That allows us to reduce the number of strategies from \(| Intr |^T \) to \( 2^T \) (intervention recommended by the decision tree plus “do nothing option”).
On top of that, to further reduce the model size we utilize specified budgeting / non-budgeting schedules. An approaches proposed by (Scheinberg and Anastasopoulos, [20]). A diagram describing the main points of this approach is shown in Figure 1. In short, the authors propose to generate the same number of strategies per asset as there are Budgeted / Not Budgeted schedules. Budgeted / Not Budgeted schedule is a 0-1 vector of size $T$. Forecast process (the one that calculates the attribute values matrix) will apply intervention according to a decision tree (given decision tree recommends an intervention) if this year is budgeted and does not if that year is not budgeted. The process repeats until asset attributes for $T$ years are computed. Note that one can use several experts (trees) to generate more strategies using the same Budgeted/Not Budgeted schedules.

![Diagram](image)

Figure 1. Methodology for strategy generation

Examples of how budgeting / non-budgeting schedules can be defined are as follows: interventions can be done only on even / odd years; interventions can be done any year; interventions can be done any year but the first; interventions can be done any year but the first and the second and so on. The decision trees that recommend ‘single’ intervention are then applied to the different budgeting / non-budgeting schedules to develop the multi-year strategy for each asset.

With this approach one uses common sense on top of assuming the existence of a decision mechanism. Common sense helps to choose a reasonable number of budgeting schedules (out of $2^T$) in order to get a representative set of strategies for the optimization to choose from.

This problem size is flexible and can be easily adjusted to get a good balance between the model’s drawbacks and its size. Scheinberg and Anastasopoulos [20] obtained very impressive results using just 10 strategies (Budgeted/Not Budgeted schedules) per asset for a 10 year plan.

5. EVALUATION OF REDUCED-SIZE TECHNIQUES

In order to analyze the performance and effectiveness of the reduced-size techniques formulated in the previous section we will consider a hypothetical asset network. In this example the asphalt road asset network has 103 sections. Current data for those sections were obtained from an existing US county database for asphalt surface and the same functional class. The only performance measure we considered in this example was the Cracking Index (100 – best score, 0 – worst score).
We used four non trivial interventions. Table 3 shows these interventions with their costs, improvements in the Cracking index and post intervention performance models. Post intervention performance models are presented in Figure 2. The decision tree (for the appropriate techniques) is shown on Figure 3.

Table 3. Treatment details

<table>
<thead>
<tr>
<th>Intervention</th>
<th>After intervention performance model</th>
<th>Unit Cost</th>
<th>Improvement / Restoration Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thick Restoration</td>
<td>3</td>
<td>28</td>
<td>set to 100</td>
</tr>
<tr>
<td>Thin Restoration</td>
<td>2</td>
<td>20</td>
<td>set to 80</td>
</tr>
<tr>
<td>Reconstruction (Flexible)</td>
<td>1</td>
<td>75</td>
<td>set to 100</td>
</tr>
<tr>
<td>Surface Coat</td>
<td>2</td>
<td>4</td>
<td>add 6</td>
</tr>
</tbody>
</table>

All of the reduced-size techniques presented in the previous section were applied to this hypothetical network. The appropriate number of variables and percent reduction in solution space size are presented in Table 4. We also considered the ‘Ranking Single-Year’ method, wherein, a solution is obtained by ranking assets by objective/constraint (usual benefit/cost) ratio. We included this trivial method just because it is often used by professionals however it is important to emphasize that this method does not solve any optimization problem.

We compared solutions for the following two possible scenarios:
- Scenario 1: Find the cheapest 5-year work plan that each year keeps the average Cracking Index above 90.
- Scenario 2: Find the cheapest 5-year work plan that each year keeps the average Cracking Index above 93.

The optimal solutions obtained for each reduction technique were compared with the optimal solution of the GAMP and the ‘optimality gaps’ are presented in Table 4. The detailed optimal cost structure for both scenarios is presented in Table 5.
The best plans are found by the solving GAMP – totally expected because this maintenance plan is an exact optimal solution. Understandably Multi-Year Single Action problem worked pretty well in 1st scenario and poorly in the second one. In order to support average index at 90 it was enough to treat each section only once in 5 years, however level 93 required either heavier interventions or several. This technique can be used for scenarios with either a low level of required performance or short time limits.

Among techniques that use decision trees the Multi-Year setup tends to produce better results than the Single-Year one. What happens is the Single-Year optimization problem tends to use cheap interventions that immediately improve conditions. The future is not taken into account. The Multi-Year approach avoids this type of misbehavior by looking at the whole time horizon and uses heavier treatments less often instead of cheap ones more often. This turns out to be better investment strategy. The difference between the two approaches increases with the analysis time limit. In our experience the Multi-Year Tree-Action technique allows savings of 10-20% of budget for the same condition constraint when compared with the Single-Year Tree-Action technique.

Table 4. Technique comparisons – optimality gap

<table>
<thead>
<tr>
<th>Technique Type</th>
<th>Problem Constraint: Minimum Cracking Index</th>
<th>Problem Size: Variables</th>
<th>Problem Size: %</th>
<th>Problem Result: Total Cost Over 5 Years</th>
<th>Problem Metric: Optimality Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SCENARIO 1</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GAMP</td>
<td>90</td>
<td>321,875</td>
<td>100.00%</td>
<td>$35,890,483</td>
<td>0.00%</td>
</tr>
<tr>
<td>Multi-Year Single Action</td>
<td>90</td>
<td>2,163</td>
<td>0.67%</td>
<td>$37,102,853</td>
<td>3.38%</td>
</tr>
<tr>
<td>Single-Year Multi-Action</td>
<td>90</td>
<td>515</td>
<td>0.16%</td>
<td>$40,119,800</td>
<td>11.78%</td>
</tr>
<tr>
<td>Multi-Year Tree-Action</td>
<td>90</td>
<td>824</td>
<td>0.26%</td>
<td>$38,565,316</td>
<td>7.45%</td>
</tr>
<tr>
<td>Single-Year Tree-Action</td>
<td>90</td>
<td>103</td>
<td>0.03%</td>
<td>$39,189,593</td>
<td>9.19%</td>
</tr>
<tr>
<td>Ranking Single-Year</td>
<td>90</td>
<td>-</td>
<td>-</td>
<td>$38,617,243</td>
<td>7.60%</td>
</tr>
<tr>
<td><strong>SCENARIO 2</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GAMP</td>
<td>93</td>
<td>321,875</td>
<td>100.00%</td>
<td>$42,558,010</td>
<td>0.00%</td>
</tr>
<tr>
<td>Multi-Year Single Action</td>
<td>93</td>
<td>2,163</td>
<td>0.67%</td>
<td>$46,650,109</td>
<td>9.62%</td>
</tr>
<tr>
<td>Single-Year Multi-Action</td>
<td>93</td>
<td>515</td>
<td>0.16%</td>
<td>$45,683,595</td>
<td>7.34%</td>
</tr>
<tr>
<td>Multi-Year Tree-Action</td>
<td>93</td>
<td>824</td>
<td>0.26%</td>
<td>$45,837,526</td>
<td>7.71%</td>
</tr>
<tr>
<td>Single-Year Tree-Action</td>
<td>93</td>
<td>103</td>
<td>0.03%</td>
<td>$46,611,051</td>
<td>9.52%</td>
</tr>
<tr>
<td>Ranking Single-Year</td>
<td>93</td>
<td>-</td>
<td>-</td>
<td>$47,247,804</td>
<td>11.02%</td>
</tr>
</tbody>
</table>
develop particular yearly maintenance plans). This period) and “detailed” stage (solve each of 5 of this paper is the direction provided to researchers to attempt to formulate
alternative and innovative asset management problems and techniques. As a very promising research in this
optimization techniques”. Such an evaluation allows researchers to discover promising optimization techniques.
The main contribution of this paper is the introduction of the concept and value of “performance evaluation of
optimization techniques”. Such an evaluation allows researchers to discover promising optimization techniques.
The second important contribution of this paper is the direction provided to researchers to attempt to formulate
alternative and innovative asset management problems and techniques. As a very promising research in this
direction authors would like to mention use of “strategy trees” (rather than single year intervention trees). Such
a decision mechanism would immediately reduce the number of strategies per asset to the number of years in
planning and would make the GAMP absolutely manageable.
Another approach worth analyzing is a 2-stage approach that would use a “general” stage (for example, solve a
20-year instance considering each 5-years as a single unit to find the best 5-year strategies for the asset within
this period) and “detailed” stage (solve each of 5-year problems taking results of “general” stage as constraints to
develop particular yearly maintenance plans).

<table>
<thead>
<tr>
<th>Technique Type</th>
<th>Plan Year 1</th>
<th>Plan Year 2</th>
<th>Plan Year 3</th>
<th>Plan Year 4</th>
<th>Plan Year 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAMP</td>
<td>$14,311,559</td>
<td>$6,141,670</td>
<td>$5,862,292</td>
<td>$4,860,978</td>
<td>$4,713,984</td>
</tr>
<tr>
<td>Multi-Year Single Action</td>
<td>$14,255,550</td>
<td>$6,134,655</td>
<td>$5,627,042</td>
<td>$5,974,087</td>
<td>$5,111,519</td>
</tr>
<tr>
<td>Single-Year Multi-Action</td>
<td>$13,869,502</td>
<td>$6,538,218</td>
<td>$6,640,269</td>
<td>$6,671,724</td>
<td>$6,400,087</td>
</tr>
<tr>
<td>Multi-Year Tree-Action</td>
<td>$14,068,115</td>
<td>$6,608,364</td>
<td>$6,623,344</td>
<td>$5,751,398</td>
<td>$5,514,095</td>
</tr>
<tr>
<td>Single-Year Tree-Action</td>
<td>$14,138,627</td>
<td>$6,458,159</td>
<td>$6,273,766</td>
<td>$6,200,522</td>
<td>$6,118,519</td>
</tr>
<tr>
<td>Ranking Single-Year</td>
<td>$14,167,914</td>
<td>$7,024,232</td>
<td>$5,620,510</td>
<td>$6,374,664</td>
<td>$5,429,923</td>
</tr>
<tr>
<td>SETUP 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GAMP</td>
<td>$21,410,666</td>
<td>$5,796,511</td>
<td>$5,571,905</td>
<td>$6,174,926</td>
<td>$3,604,002</td>
</tr>
<tr>
<td>Multi-Year Single Action</td>
<td>$21,598,832</td>
<td>$6,305,756</td>
<td>$6,156,225</td>
<td>$6,241,580</td>
<td>$6,347,716</td>
</tr>
<tr>
<td>Single-Year Multi-Action</td>
<td>$21,203,944</td>
<td>$5,699,723</td>
<td>$6,148,538</td>
<td>$6,372,016</td>
<td>$6,259,374</td>
</tr>
<tr>
<td>Multi-Year Tree-Action</td>
<td>$21,805,919</td>
<td>$6,389,392</td>
<td>$6,013,287</td>
<td>$6,047,050</td>
<td>$5,581,878</td>
</tr>
<tr>
<td>Single-Year Tree-Action</td>
<td>$21,791,558</td>
<td>$6,484,629</td>
<td>$6,092,136</td>
<td>$6,006,669</td>
<td>$6,236,059</td>
</tr>
<tr>
<td>Ranking Single-Year</td>
<td>$22,114,440</td>
<td>$8,169,780</td>
<td>$4,749,466</td>
<td>$6,011,232</td>
<td>$6,202,886</td>
</tr>
</tbody>
</table>

6. CONCLUSIONS

The GAMP is one of the most important problems of Asset Management. Their main goal is to produce
intervention plans. There are numerous attempts by transportation agencies to approximate GAMP solutions in
order to provide better quality service and highway security to the public. Unfortunately the problem size grows
rapidly and in practice the resulting instances are too big and hard to solve. This results in adaptation of simple
ranking or sorting algorithms by many providers and therefore some agencies are still using a manager to do
maintenance planning. To address the issue four techniques that can be used to reduce GAMP size were
presented and their performance analyzed. Using numerical examples, the authors showed that additional model
assumptions which reduce model size might work in some cases but the performance will degrade if the
assumption is not valid. A more robust approach is the use of a decision mechanism. The following presents a
short summary of our findings:

- Multi-Year models provide better solutions – much better solutions for longer time horizons
- The use of decision trees to reduce model size works fairly well.
- Decision trees provide uniform logic for decision making. This makes understanding of the solution easy
  which boosts engineer confidence.

The main contribution of this paper is the introduction of the concept and value of “performance evaluation of
optimization techniques”. Such an evaluation allows researchers to discover promising optimization techniques.
REFERENCES:


