Efficient Long-Term Degradation Profiling in Time Series for Complex Physical Systems

Liudmila Ulanova *
UC Riverside
lulan001@ucr.edu

Tan Yan
NEC Laboratories America
yan@nec-labs.com

Haifeng Chen
NEC Laboratories America
haifeng@nec-labs.com

Guofei Jiang
NEC Laboratories America
gfj@nec-labs.com

Eamonn Keogh
UC Riverside
eamonn@cs.ucr.edu

Kai Zhang
NEC Laboratories America
kzhang@nec-labs.com

ABSTRACT

The long term operation of physical systems inevitably leads to their wearing out, and may cause degradations in performance or the unexpected failure of the entire system. To reduce the possibility of such unanticipated failures, the system must be monitored for tell-tale symptoms of degradation that are suggestive of imminent failure. In this work, we introduce a novel time series analysis technique that allows the decomposition of the time series into trend and fluctuation components, providing the monitoring software with actionable information about the changes of the system's behavior over time. We analyze the underlying problem and formulate it to a Quadratic Programming (QP) problem that can be solved with existing QP-solvers. However, when the profiling resolution is high, as generally required by real-world applications, such a decomposition becomes intractable to general QP-solvers. To speed up the problem solving, we further transform the problem and present a novel QP formulation, Non-negative QP, for the problem and demonstrate a tractable solution that bypasses the use of slow general QP-solvers. We demonstrate our ideas on both synthetic and real datasets, showing that our method allows us to accurately extract the degradation phenomenon of time series. We further demonstrate the generality of our ideas by applying them beyond classic machine prognostics to problems in identifying the influence of news events on currency exchange rates and stock prices. We fully implement our profiling system and deploy it into several physical systems, such as chemical plants and nuclear power plants, and it greatly helps detect the degradation phenomenon, and diagnose the corresponding components.

1. INTRODUCTION

The long term operation of physical systems causes degradation of their components. Degradations, such as wearing out or aging [1], lead to condition changes expressed in deviations in mean values,

---

*Work done during an internship at NEC Laboratories America, Princeton.

[1]We use terms *degradation* and *aging* interchangeably in this paper.

amplitudes or frequencies in the sensed signals of the system, and are an unavoidable property of any system. However, well-timed detection and profiling of an aging phenomenon or trend help avoid consequences and problems it may cause and are crucial tasks in the modern machinery. It is important to note, if one component of a system fails, it may lead to the stoppage of the whole system, and, as already well known, even short interruption in production process may cause huge money waste and serious business related issues [9].

The problem of trend analysis and extraction in time series has attracted significant attention recently, and a plethora of methods have been designed [4,14]. However, the proposed schemes mainly focus on extracting general trend behavior from time series, and none of them specifically addresses the aging detection and profiling without any prior knowledge of the time series properties (i.e., seasonality, level of noise, etc.).

The *aging phenomenon* profiling is a critical task in detection and analysis of the degradation behavior in time series obtained from sensors on machinery. Such a problem is complicated because a system is usually operated with strong patterns. The aging behavior is usually invisible due to high noise and operational signals, and thus is too small to be detected. Figure 1 shows an example of a one-year-long time series collected from a pipe of a chemical plant. A component inside the pipe started wearing out at the beginning of January 2013, and another started at the beginning of March. This leads to the degradation of the whole pipe. Such aging phenomenon is too tiny to be captured by human eyes but may cause damage to the whole system.

Figure 1: Example of time series showing aging (long-term degradation) phenomenon.

To resolve this issue, in this paper, we aim to provide a time series analysis technique that accurately extracts the aging phenomena of given time series, analyzes the aging behavior, and ranks its severity. We observe that in physical systems, the aging usually is generated in long-term and incremental fashions, such as buildups or component wear out, and thus, naturally, the aging behavior

can be modeled as a monotonic signal. Moreover, for systems under normal operations, the long-term operational signal should be stationary. For example, it does not make sense if a system is normally operated to always increase/decrease temperature, water flow, or pressure for five years, although the instant value can fluctuate at different time points. Under such observations, we want to develop an aging profiling engine that decomposes the given time series into two components - a *monotonic aging trend* that captures all the degradation phenomena in the time series and a *fluctuation term* that represents the normal operational signals and noises. In decomposition, we analyze the aging trend to profile its changes and evolutions including the slope and the shape that indicates the severity and starting point of the degradation.

More particularly, we formulate the aging extraction problem as an optimization problem. Its objective function consists of two parts: (1) Minimizing the reconstruction error to ensure an accurate profiling, and (2) Ensuring flatness of the fluctuation term over time so that the aging trend captures as much degradation signal as it can. Under such objectives, the solution must also satisfy two constraints: (1) the extracted aging trend needs to be monotonic to satisfy the nature of the component wearing out, and (2) all the extracted signals need to be real valued and non-negative. We analyze the formulated objective function and convert this problem statement to a Quadratic Programming (QP) formulation [11] that can be solved with existing well-known solvers. However, when the profiling resolution is high, as generally required by real-world applications, such a problem becomes intractable to general QP-solvers. To further speed up the problem solving, we propose a matrix transformation technique to simplify the constraints and reduce the number of constraints from two to one – only non-negativity constraint. We convert the problem to a Non-negative QP problem and solve it much more efficiently using iterative updates [21]. We show that, using the optimization techniques proposed in this paper, the solution of the problem is tractable, and thus applicable to large-scale time series profiling.

We design two scoring functions to give an aging score and a confidence score to analyze the extracted aging component of the time series. Based on such scores, our aging profiling engine reports the result of decomposition with a ranking of the aging severity and the corresponding confidence level of the extraction, which can assist the domain experts to further diagnose the system.

We demonstrate our ideas on both synthetic and real data, showing that our method allows us to accurately divorce the trend and the shape of rapidly arriving time series. We further demonstrate the generality of our ideas by applying them beyond classic machine prognostics to problems in identifying the influence of news events on currency exchange rates. We fully implement our aging profiling system and deploy it into several physical systems, such as a chemical plant and a nuclear power plant in Japan, which greatly helps detect the aging phenomenon and diagnose the corresponding components.

To summarize, in this paper we make the following three contributions:

- We formulate an objective function for extracting trend and fluctuation terms and build a model to transit the aging profiling requirements into mathematical formulation;

- We demonstrate that the problem is a convex problem, convert it to a QP formulation, and provide a fast optimization technique to solve the problem with Non-negative QP formulation;

- We fully develop the system and deploy it into several real-world complex systems.

The rest of the paper is organized as follows. In Section 2 we provide our detailed scheme and problem formulation. Section 3 gives optimization techniques to efficiently solve our problem. Section 4 presents the conducted experiments. The related work is presented in Section 5. We conclude this paper in Section 6.

2. THE AGING PROFILING ALGORITHM

In this section, we describe our approach for aging profiling in time series data. First, we present the formal definition of the problem. Then, we form the objective function and formulate it to a Quadratic Programming Problem, a subclass of the convex problems whose optimal solution can be obtained with existing solvers. After that, we present the scoring functions to rank the aging severity of the input time series and the confidence level of out system.

2.1 Problem Statement

In physical and other complex systems, the monitored signals (time series) generally are a mixture of aging phenomena, system operational signals, and noises. Our goal is to extract and profile the aging components/phenomena/signals out of the given time series. The aging extraction problem can be defined as follows: given an input time series of length T, $\mathbf{s} \in \mathbf{R}^{T \times 1}$, we want to decompose it into the following terms:

$$\mathbf{s} = \mathbf{x}_a + \mathbf{x}_f \qquad (1)$$

where $\mathbf{x}_a$ is the aging term and $\mathbf{x}_f$ is the fluctuation term. Illustration of the desired decomposition is presented in Figure 2.
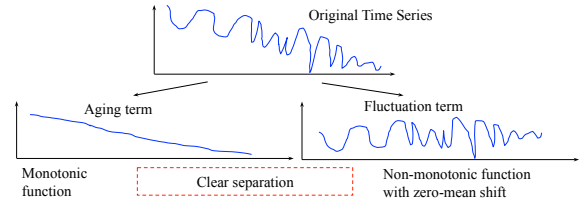

Figure 2: The desired decomposition of a time series to extract the aging trend: the clear separation of the two components – the aging trend and the fluctuation term.

We require the trend component to be monotonic – either non-increasing or non-decreasing [2], and the fluctuation term to be flat, i.e., stationary, to make sure it does not interfere with the aging trend. Having this problem statement, we formulate our problem into an objective function optimization in the following section.

2.2 Quadratic Programming Formulation

To ensure that the trend $\mathbf{x}_a$ is monotonic, we can simply write the constraint of the objective function as

$$\mathbf{x}_a(t) \leq (or \geq)\mathbf{x}_a(t+1). \qquad (2)$$

To make the fluctuation term stationary and do not have the mean change over the time, we segment the time series, and ensure the mean value difference of each segment to be as small as possible. More specifically, given a time series, we partition it into K segments, each having mean value m_k, $k = 1, 2, ..., K$. Then, we minimize the difference between mean values of each pair of the segments to ensure flatness of the fluctuation term over time as:

$$\min \sum_{i,j=1}^{K} (m_i - m_j)^2 \mathbf{W}_{ij} \qquad (3)$$

[2] For other aging behaviors, such as frequency or variance change, they can always be transformed to value change and modeled as monotonic signals.

Here $\mathbf{W}$ is a weight matrix, each entry $\mathbf{W}_{ij}$ of which specifies the weight we want to assign in order to minimize the mean difference between the segment pair m_i and m_j. If $\mathbf{W}$ is set to be all 1's, then all the segments have equal weight. In practice, to ensure the long-term mean change being minimized, we can increase weights when the distance of the segment pair increases (e.g., linear growth of the weight based on the distance between segments m_i and m_j). In this way, the farther the two segments are apart from each other, the more weight is given to minimize their mean difference.

We formulate the objective function of the optimization problem as minimizing the reconstruction error $\|\mathbf{s} - \mathbf{x}_a - \mathbf{x}_f\|^2$ and ensuring flatness of the fluctuation term over time (Equation (3)), which is subject to the following constraints: (1) monotonicity constraint and (2) non-negativity constraint. The monotonicity constraint can be represented using Equation (2). The non-negativity constraint means that the extracted aging trend and fluctuation term must have only non-negative values. It assumes that the original time series has only non-negative values as well, but if it is not the case it can be easily shifted up by summing up the original values with the absolute value of the minimum (negative) value of the time series. Combining this with Equations (2) and (3), the whole objective function can be written the following way:

$$
\begin{aligned}
\min \quad & \|\mathbf{s} - \mathbf{x}_a - \mathbf{x}_f\|^2 + \sum_{i,j=1}^{K} (m_i - m_j)^2 \mathbf{W}_{ij} \\
s.t. \quad & \mathbf{x}_a(t) \le (or \ge) \mathbf{x}_a(t+1), \ \forall_{t=1,2,...,T-1} \\
& \mathbf{x}_a \ge 0, \mathbf{x}_f \ge 0
\end{aligned} \tag{4}
$$

However, Equation (4) cannot be solved in the form presented above. We now transform it to a Quadratic Programming formulation whose optimal solution can be obtained. We begin with rewriting Equation (3), the flatness objective. For each of the segments m_i, we define an indicator vector $\mathbf{e}_i$ as follows:

$$
\mathbf{e}_i = [\underbrace{0,0,...,0}_{1st\ seg}, \ \underbrace{0,0,...,0}_{2nd\ seg}, \ ..., \underbrace{\frac{1}{l}, \frac{1}{l}, ..., \frac{1}{l}}_{ith\ seg}, \underbrace{0,0,...,0}_{Kth\ seg}, \ ]^\top, \tag{5}
$$

where l is the length of the segment. We let

$$
\mathbf{E} = [\mathbf{e}_1, \ \mathbf{e}_2, ..., \mathbf{e}_K]^\top,
$$

and then such flatness objective (Equation (3)) can be written as

$$
\min \mathbf{x}_f^\top \left(\mathbf{E}^\top \mathbf{L} \mathbf{E} \right) \mathbf{x}_f \tag{6}
$$

where $\mathbf{L} = \mathbf{D} - \mathbf{W}$ and $\mathbf{D} = diag(\mathbf{W} \cdot \mathbf{1})$.

After that, in Equation (4), we replace the original form of the flatness objective with its new form in Equation (6) and obtain the following formulation:

$$
\begin{aligned}
\min \quad & \|\mathbf{s} - \mathbf{x}_a - \mathbf{x}_f\|^2 + \lambda \cdot \mathbf{x}_f^\top \left(\mathbf{E}^\top \mathbf{L} \mathbf{E} \right) \mathbf{x}_f \\
s.t. \quad & \mathbf{x}_a(t) \le \mathbf{x}_a(t+1), \ \forall_{t=1,2,...,T-1} \\
& \mathbf{x}_a \ge 0, \mathbf{x}_f \ge 0.
\end{aligned} \tag{7}
$$

Then, we rewrite Equation (2), the monotonicity constraint. We define $\mathbf{x} = [\mathbf{x}_a^\top, \mathbf{x}_f^\top]^\top$, $\mathbf{E}_2 = [\mathbf{I}, \ \mathbf{I}]$, $\mathbf{E}_1 = [\mathbf{0}, \ \mathbf{I}]$, where $\mathbf{I}$ is a $T \times T$ identity matrix. We define matrices $\mathbf{B}$ and $\mathbf{C}$ as follows:

$$
\mathbf{B} = \begin{bmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & & .. & \\ & & & 1 & -1 \end{bmatrix}, \mathbf{C} = [\mathbf{B}, \ \mathbf{0}_{T-1 \times T-1}]
$$

Under such a definition, the monotonicity constraint can be written as

$$
\mathbf{C}\mathbf{x} \ge 0, \tag{8}
$$

and the non-negative constraints can be written as

$$
\mathbf{x} \ge 0. \tag{9}
$$

We further rewrite the objective function in Equation (7) and obtain its form as follows:

$$
\begin{aligned}
\min \quad & \mathbf{x}^\top \left(\mathbf{E}_2^\top \mathbf{E}_2 + \lambda \mathbf{E}_1^\top \mathbf{E}^\top \mathbf{L} \mathbf{E} \mathbf{E}_1 \right) \mathbf{x} - 2(\mathbf{E}_2^\top \mathbf{s})^\top \mathbf{x} \\
s.t. \quad & \mathbf{C}\mathbf{x} \ge 0 \\
& \mathbf{x} \ge 0
\end{aligned} \tag{10}
$$

Now, the Equation (10) has a form of a Quadratic Programming optimization problem. Later in Section 3 we propose an efficient optimization method to solve this problem.

2.3 Scores of Aging Severity and Confidence

We decompose a given time series into the aging part (trend) and the fluctuation part. During the decomposition, the reconstruction error is introduced, and thus, the confidence of such decomposition must depend on the value of the error. We use scores to assess the severity of the aging as well as the confidence our tool provides. Essentially, the aging score can be described by the slope of the aging trend, while the confidence score depends on the closeness of the slope of the extracted aging trend to the slope of the original time series. We tune two sigmoid functions to quantify the two scores. For the aging score, we use the slope of the extracted aging trend as the input of the sigmoid function, that gives a value from 0 (no aging expressed) to 1 (severe aging). For confidence score, we use the difference ratio between the slope of the original time series and the slope of the extracted aging trend, i.e., $(ori.\_slope - ext.\_slope)/ori.\_slope$, as the input. The higher the ratio is, the lower the confidence score will be. The two scoring functions can be flexibly tuned according to different application requirements. For example, for five-year monitoring of a physical system, we can pre-tune aging score to give 0 for zero-slope aging trend, while giving 1 when the slope is greater than 0.015, as shown in Figure 3(a). The confidence score function can be tuned so that it scores 1 when the ratio is 0, while it scores 0 when the ratio is close to 1, as shown in Figure 3(b).
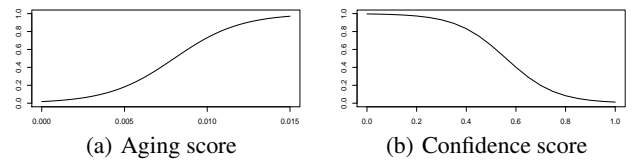

(a) Aging score   (b) Confidence score

Figure 3: Aging score and confidence score sigmoid functions

To summarize, the design of the aging profiling system is illustrated in Figure 4.

3. SOLVING PROBLEM (10) EFFICIENTLY

We formulate the underlying aging profiling problem to a QP problem. However, the general QP-solvers are used for a wide range of purposes and are considerably slow, especially for large time series (e.g., more than 1000 data points). Essentially, finding a solution for a time series of 1000 data points is equivalent to
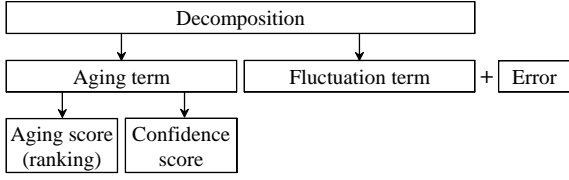
Figure 4: The system overview. A time series is decomposed into the aging and fluctuation terms. Based on the aging term we obtain the aging score (ranking) and the confidence score for it.

Figure 5: Aging trend x_a and its difference values $\bar{x}_a$

solving a system of equations with 1000 variables, which is very computationally costly, and thus, is not tractable.

In this section, we aim to reduce the computational cost and provide a tractable solution for reasonably long time series. Our idea is to further rewrite the QP formulation in Equation (10) to eliminate the monotonicity constraint and only keep the non-negativity constraint to make the problem much easier to solve.

3.1 Non-negative Formulation

We observe that *monotonicity constraint expresses the fact that the difference between two neighboring elements is non-negative* [3]. Based on the observation, we introduce a transition variable $\bar{\mathbf{x}}_a$, and change the problem statement to the following way: instead of using $\mathbf{x}_a$ as a part of the variables, we can use the following surrogate $\bar{\mathbf{x}}_a$ as the variable, which satisfies the equality

$$\mathbf{x}_a(1) = \bar{\mathbf{x}}_a(1)$$
$$\mathbf{x}_a(2) = \bar{\mathbf{x}}_a(1) + \bar{\mathbf{x}}_a(2)$$
$$...$$
$$\mathbf{x}_a(k) = \bar{\mathbf{x}}_a(1) + \bar{\mathbf{x}}_a(2) + ... + \bar{\mathbf{x}}_a(k).$$

In other words, the first element in $\bar{\mathbf{x}}_a$ is equal to the first element of $\mathbf{x}_a$, and the remaining elements in $\bar{\mathbf{x}}_a$ represent the difference between two neighboring elements of $\mathbf{x}_a$.

The relationship between x_a and $\bar{x}_a$ can be written in the following way:

$$\mathbf{x}_a = \Delta \bar{\mathbf{x}}_a \tag{11}$$

where $\Delta = \Delta^-$ if the trend is nonincreasing, while $\Delta = \Delta^+$ if the trend is nondecreasing

$$\Delta^- = \begin{bmatrix} 1 & & & & \\ 1 & 1 & & & \\ 1 & 1 & 1 & & \\ ... & & & & \\ 1 & 1 & 1 & ... & 1 \end{bmatrix} \tag{12}$$

$$\Delta^+ = \begin{bmatrix} 1 & & & & \\ 1 & -1 & & & \\ 1 & -1 & -1 & & \\ ... & & & & \\ 1 & -1 & -1 & ... & -1 \end{bmatrix} \tag{13}$$

An example of a trend x_a and its difference values $\bar{x}_a$ is illustrated in Figure 5.

We let $\bar{\mathbf{x}} = [\bar{\mathbf{x}}_a^\top \ \mathbf{x}_f^\top]^\top$. By using this transform, we can merge the monotonicity constraint $\mathbf{Cx} \geq 0$ in Equation (8) and non-

negative constraint $\mathbf{x} \geq 0$ in Equation (9) to just one constraint:

$$\bar{\mathbf{x}} \geq 0. \tag{14}$$

We let $\bar{\mathbf{E}}_2 = [\Delta \ \mathbf{I}]$. The original QP problem in Equation (10) can be written in the following form with the only one constraint:

$$\min \quad \bar{\mathbf{x}}^\top \left(\bar{\mathbf{E}}_2^\top \bar{\mathbf{E}}_2 + \lambda \mathbf{E}_1^\top \mathbf{E}^\top \mathbf{LEE}_1 \right) \bar{\mathbf{x}} - 2(\bar{\mathbf{E}}_2^\top \mathbf{s})^\top \bar{\mathbf{x}}$$
$$s.t. \qquad\qquad\qquad\qquad \bar{\mathbf{x}} \geq 0 \tag{15}$$

We further define

$$\mathbf{Q} = \left(\bar{\mathbf{E}}_2^\top \bar{\mathbf{E}}_2 + \lambda \mathbf{E}_1^\top \mathbf{E}^\top \mathbf{LEE}_1 \right)$$
$$\mathbf{c} = \bar{\mathbf{E}}_2^\top \mathbf{s}, \tag{16}$$

then the problem can be written as a simple *non-negative* QP problem:

$$\min \quad \bar{\mathbf{x}}^\top \mathbf{Q} \bar{\mathbf{x}} - 2\mathbf{c}^\top \bar{\mathbf{x}}$$
$$s.t. \qquad \bar{\mathbf{x}} \geq 0 \tag{17}$$

After solving Equation (17), the obtained aging trend $\mathbf{x}_a$ can be recovered with $\bar{\mathbf{x}}_a$ as it is shown in E. q. (11). The next section shows that indeed such a non-negative QP problem can be solved much more efficiently than a general QP problem.

3.2 Avoiding General QP-solver

The advantage of using the non-negative QP formulation is that it makes the structure simpler, since non-negativity constraints are much easier to handle than linear inequality constraints. Typically they lead to closed-form iterations that can be computed very efficiently. For example, the multiplicative update studied in [21] can be used to obtain the solution. Let $\mathbf{Q}^+$ and $\mathbf{Q}^-$ be defined as

$$\mathbf{Q}_{ij}^+ = \begin{cases} \mathbf{Q}_{ij}, & \mathbf{Q}_{ij} \geq 0 \\ 0, & otherwise \end{cases} \quad \mathbf{Q}_{ij}^- = \begin{cases} |\mathbf{Q}_{ij}|, & \mathbf{Q}_{ij} < 0 \\ 0, & otherwise \end{cases} \tag{18}$$

Then, the update can be written as

$$\bar{\mathbf{x}}_{i+1} \leftarrow \bar{\mathbf{x}}_i \left[\frac{-\mathbf{c}_i + \sqrt{\mathbf{c}_i^2 + 4(\mathbf{Q}^+\bar{\mathbf{x}})_i(\mathbf{Q}^-\bar{\mathbf{x}})_i}}{(2\mathbf{Q}^+\bar{\mathbf{x}})_i} \right], \tag{19}$$

where $\mathbf{x}_i$ is the vector $\mathbf{x}$ obtained after i iterations.

A good initialization can improve convergence. One can use $|(\mathbf{Q}^{-1}\mathbf{c})|$ suggested in [21] as the starting solution. To compute $\mathbf{Q}^{-1}$, one can use low-rank approximation techniques which is linear complexity.

This avoids using general QP-solvers that are typically designed for general purpose formulations with both linear and nonlinear constraints, and thus, makes the solving process much more efficient. A comprehensive comparison between solving efficiency using general QP-solvers and the proposed method is later shown in Section 4.1.3.

3.3 Formal Algorithm Definition

The formal presentation of our algorithm for time series decomposition can be found in Algorithm 1.

---

[3]In case of nonincreasing sequence every preceding element is greater or equal to the following, and in case of nondecreasing sequence, vice versa, each following element is greater or equal to the preceding one.
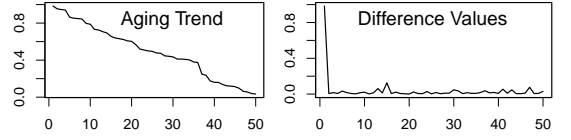
Algorithm 1 The aging profiling algorithm

Notations:
  s: intput time series of lenght T
  x_a: aging trend (array of length T)
  x_f: fluctuation term (array of length T)
  $\bar{x}_a$: difference values of aging trend x_a
  x: concatenation of aging trend and fluctuation term, $x = x_a + x_f$
  $\bar{x}$: concatenation of $\bar{x}_a$ and $\mathbf{x}_f$
  Δ: transform matrix from $\bar{x}_a$ to x_a
  ϵ: small value used as an algorithm stopping criterion
1: Check the slope of s
2: **if** slope is negative **then**
3:     $\Delta = \Delta^-$
4: **else**
5:     $\Delta = \Delta^+$
6: **end if**
7: Compute Q, c
8: $\bar{x}_a = |(\mathbf{Q}^{-1}\mathbf{c})|$
9: $f_{old} = 0, difference = \text{Infinity}$
10: **while** $difference > \epsilon$ **do**
11:     Compute $\bar{x}$ by (19)
12:     $f_{new} = \bar{x}^\top \mathbf{Q}\bar{x} - 2\mathbf{c}^\top \bar{x}$
13:     $difference = |(f_{old} - f_{new})|$
14:     $f_{old} = f_{new}$
15: **end while**
16: $\bar{x}_a = \bar{x}[1:T], x_f = \bar{x}[(T+1):2T]$
17: $x_a = \Delta \bar{x}_a$
18: **return** x_a, x_f

In the algorithm, the first step is to check the slope of the input time series and decide if Δ^+ or Δ^- should be used. Then, the next step in line 7 is to compute $\mathbf{Q}$ and $\mathbf{c}$ according to the Equation (16). In line 8, we assign initial value of $\bar{x}_a$ to be $|(\mathbf{Q}^{-1}\mathbf{c})|$. After that, we execute the iterative updates until the difference between objective function values becomes less than ϵ, e.g., 0.001. In line 16, we split the $\bar{x}_a$ into its two components - the aging trend difference values and the fluctuation term, and recover the aging trend by multiplying Δ by $\bar{x}_a$ in line 17. The algorithm returns the extracted aging trend and the fluctuation term.

4. EXPERIMENTS

We have fully developed our aging profiling system and deployed it into several physical systems, such as chemical plants and nuclear power plants in Japan. In this section, we first evaluate the aging profiling accuracy, computational complexity, and the intuition of the parameter tunning of the scheme by applying it to synthetic datasets with different properties. Then, we demonstrate the good profiling and diagnosis performance by applying it to the real-world data collected from one of the systems it is deployed in. Finally, we further demonstrate the generality of our scheme by applying it beyond classic machine prognostics to problems in identifying the influence of news events on currency exchange rates and stock prices.

4.1 Synthetic Data with Different Properties

We generate synthetic data that covers time series with different properties in trend, seasonality, and noise. For trend component, we generate three types of aging trends: no trend, linear trend, and nonlinear trend. Figure 6 summarizes trends that are used in the synthetic data generation. Essentially, to generate a linear trend, we applied the slope in the whole length of time series. For a nonlinear trend, its first half is a flat line (no trend), and the non-linear curve starts in the middle of the time series. We intentionally generate the nonlinear trend this way to test whether our scheme can detect the starting point of the aging. Moreover, all the generated trends are

very small (can be seen from their slope values) in order to ensure that the aging trend is not obvious or visible by eyes.
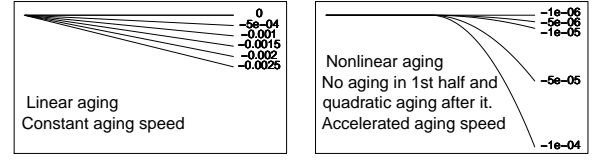

Figure 6: Trends used in synthetic data generation

The fluctuation terms are built according to the following scheme:

- sine wave: $\sin(a\pi t), a = [1/16, 1/8, 1/4, 1/2, 1]$,

- noise: $b * rand(0,1), b = [1/4, 1/2, 1, 2, 4]$,

- combination of noise and sine wave: $\sin(a\pi t) + b * rand(0,1)$.

Combinations of possible trends and fluctuations give nine families of time series, presented in Figure 7. In total, we generated 385 synthetic time series, each containing 5000 points. The next section explains parameter setup for the experiments and their justification.
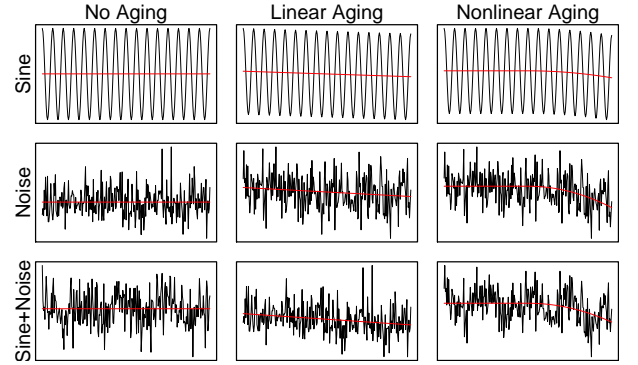

Figure 7: Synthetic Data Examples

4.1.1 Robustness to Different Parameter Values

As a careful reader could notice, our model has several parameters, namely λ, K, and $\mathbf{W}$. λ balances the minimization weights between reconstruction error and flatness of the fluctuation term. K control the resolution of the flatness of the fluctuation term. $\mathbf{W}$ adjusts the weights in the fluctuation term and can be set to linearly increase depending on the distance between the pairwised segments (e.g., $1, 2, 3, \ldots$).

To evaluate the effect of different parameter values, we list a range of values for each parameter and try all the combinations to all the synthetic time series. In Table 1, for each time series, we record the best and worst profiling accuracies that are the minimum and the maximum values of the distances between the estimated and actual trends normalized by the time series length. As we can observe, the difference between the reconstruction results is not significant. Thus, our system is not sensitive to the values of the parameters. In practice, based on our extensive experiments, we suggest to set $\lambda = 0.1$ and $K = 5$.

Table 1: Best and Worst Reconstruction Results

Family	Slope (C)	Best	Worst	Difference
Linear	0	0	0.0094	0.0094
Linear	5.0E-04	0.0005	0.0081	0.0077
Linear	1.0E-03	0.0007	0.0076	0.0068
Linear	1.5E-03	0.0009	0.0077	0.0068
Linear	2.0E-03	0.0011	0.0093	0.0082
Linear	2.5E-03	0.0014	0.0116	0.0102
Quadratic	4.8E-05	0.0002	0.0095	0.0092
Quadratic	2.4E-04	0.0009	0.0093	0.0083
Quadratic	4.8E-04	0.0015	0.0090	0.0075
Quadratic	2.4E-03	0.0040	0.0135	0.0095
Quadratic	4.8E-03	0.0048	0.0183	0.0135

4.1.2 Aging Profiling Accuracy

To evaluate the decomposition quality, we compute the Euclidean distances between the extracted and actual trends. We compare our method with Singular Spectrum Analysis (SSA), Moving Average (MA), and Wavelet. Table 2 presents the results of the comparison.

Table 2: Euclidean distance between actual and extracted trends

Family	Slope (C)	Ours	SSA	SMA	Wavelets
Linear	0	**0.0004**	0.0036	0.0158	0.0112
Linear	5.0E-04	**0.0005**	0.0034	0.0157	0.0112
Linear	1.0E-03	**0.0008**	0.0034	0.0160	0.0115
Linear	1.5E-03	**0.0010**	0.0034	0.0160	0.0116
Linear	2.0E-03	**0.0012**	0.0036	0.0158	0.0117
Linear	2.5E-03	**0.0015**	0.0035	0.0157	0.0119
Quadratic	4.8E-05	**0.0004**	0.0033	0.0157	0.0111
Quadratic	2.4E-04	**0.0010**	0.0035	0.0158	0.0111
Quadratic	4.8E-04	**0.0017**	0.0037	0.0158	0.0111
Quadratic	2.4E-03	0.0070	**0.0055**	0.0159	0.0124
Quadratic	4.8E-03	0.0099	**0.0068**	0.0161	0.0146

From this table we can see that our method performs best among the majority of the compared schemes with the smallest error between the extracted trend and the ground-truth trend. Especially when the slope is small and the aging phenomenon is tiny, our method performs about 10 times better SSA and 30 betters than SMA and Wavelets. Moreover, although in two cases our method gets higher error than SSA, the aging trend extracted by our scheme is monotonic and satisfies the nature of the aging behavior, while SSA, SMA or Wavelets does not have such advantage as they have no monotonic constraint in extracting the trend. A comparison of extracted slopes with our method and actual trends is shown in Figure 8, which demonstrates how the extracted trends correspond to the actual trends.

In addition, Figure 9 demonstrates an example of outputs from our tool. From this figure, we can see our profiling scheme accurately detects and profiles the aging shape of the input time series.

4.1.3 Computational Complexity

Any algorithm that utilizes iterative updates opens a question on how fast it converges upon a solution. This answer depends on the choice of ϵ that satisfies the accuracy of the results and gives a reasonable speed of computation. For all our experiments we fixed ϵ value to be 0.001, and set the initial values for the solution to be $|(\mathbf{Q}^{-1}\mathbf{c})|$. Figure 10 shows the average converge speed for time
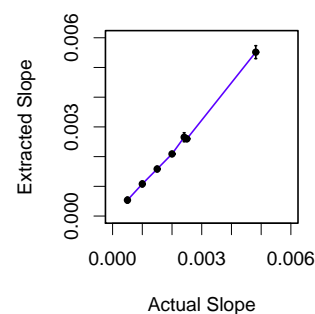

Figure 8: The slopes of the actual and extracted trends

series with 300 data points. From this figure, we can see that the algorithm converges within about 60 iterations.
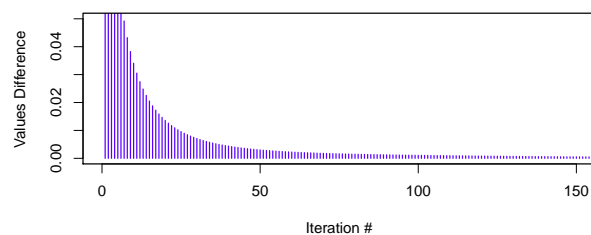

Figure 10: Convergence speed for time series length of 300 data points

To compare the efficiency of the original formulation solved with QP-solvers and the optimized non-negative QP formulation with iterative updates (named *No QP-solver*), we run load tests for different length of time series and plot the execution time for time series with length from 500 to 5000 in Figure 11. The running time is based on a PC with CPU i7-920 3.0GHz with 8 cores [4]. From this figure, it is obvious to see that our approach outperforms general QP-solver based technique, and its speed is about 30 times faster than general QP-solvers. Please note that, since our goal is to capture long-term degradations in physical systems, we do not need a very fine granularity and usually 2000 points is enough for 5-years of monitoring data.
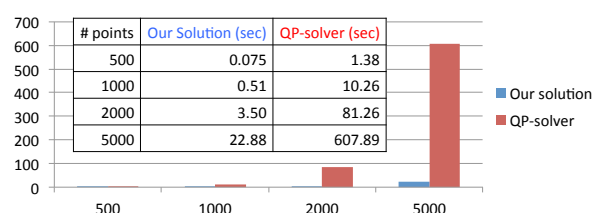

Figure 11: Solution time for QP-solver and without QP-solver

4.1.4 Aging Trend Ranking and Confidence Score

We use sigmoid functions to define the aging and confidence scores shown in Figure 3(a) and (b). The aging scores of time series with different slopes of trends are shown in Figure 12. From this figure, we can see that the aging score computed based on the

---

[4]The algorithm can be easily parallelized along multiple time series, as there is no need of memory sharing between different time series.

(a) No Aging, Sine   (b) Linear Aging, Sine   (c) Noninear Aging, Sine

(d) No Aging, Noise   (e) Linear Aging, Noise   (f) Nonlinear Aging, Noise

(g) No Aging, Sine+Noise   (h) Linear Aging, Sine+Noise   (i) Nonlinear Aging, Sine+Noise
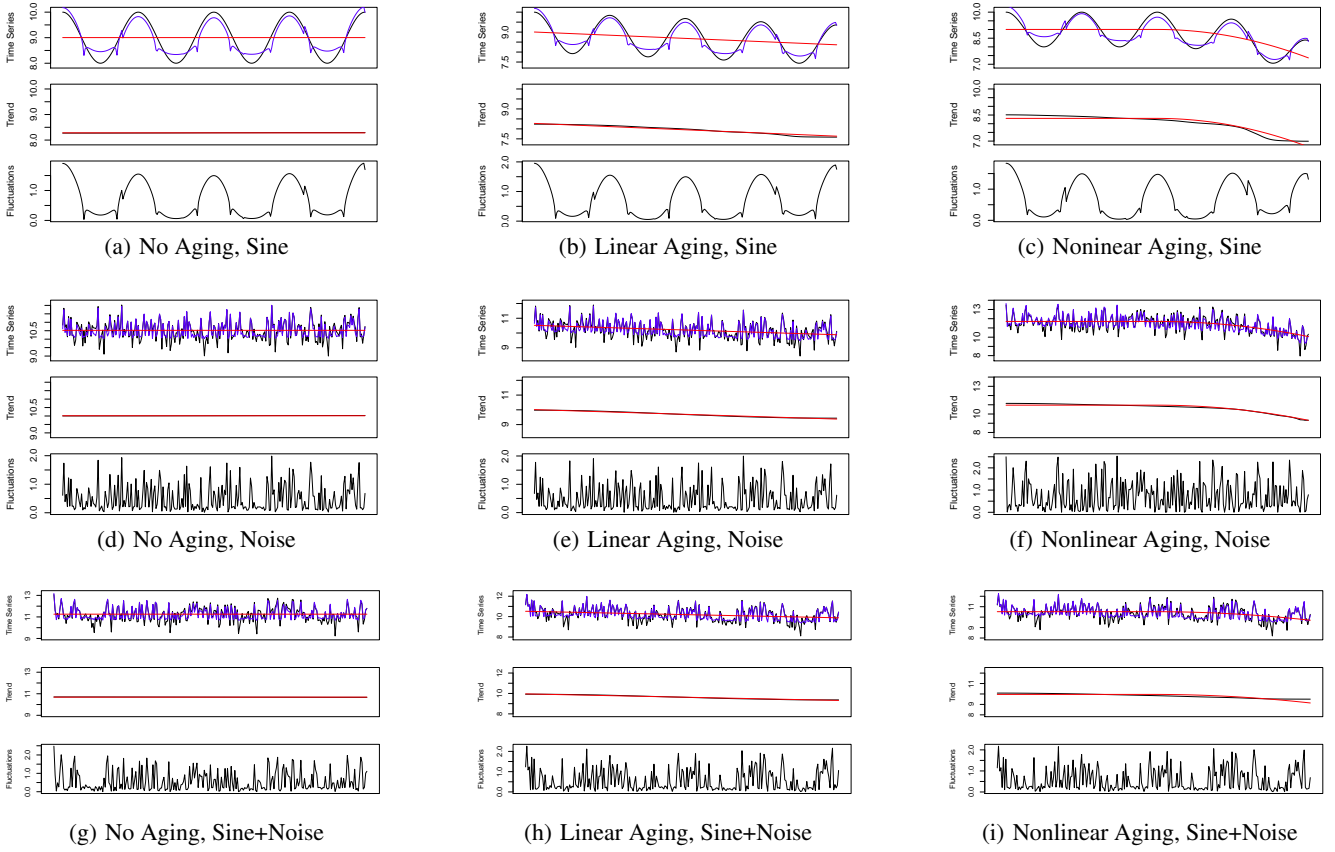
Figure 9: The examples of graphical output of our tool. The left column is a set of time series without aging; the center column is a set of time series with linear trends; the right column is for the time series with quadratic trends starting from the middle. In each plot, the first row shows the original time series (in black), its reconstructed curve, and its actual trend (in red). The second row shows the actual (in red) and extracted (in black). The bottom row shows the reconstruction error of the whole time series.

extracted aging trend highly correlates with the slope value of the actual aging trend.
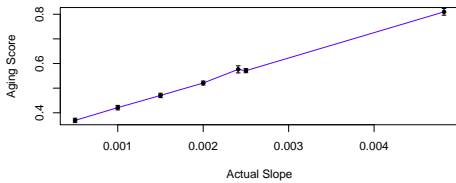

Figure 12: Aging Score Values

Figure 13: Confidence Score Values

Similarly, the confidence scores presented in Figure 13 show the closeness of the slopes of the original time series and the extracted trend. From this figure we can see that when the slope of the actual trend is greater than 0.001, our scheme can have a confidence score higher than 0.99.

4.2 Case Studies with Real-World Data

In this section, we evaluate the profiling performance of our method using real-world datasets collected from different markets: (1) multi-variant time series collected from sensors in a manufacture system of a chemical plant, (2) time series from current exchange rate between Russian Rubles and Ukrainian Hryvnias, and (3) time series from the US stock price of Apple Inc.
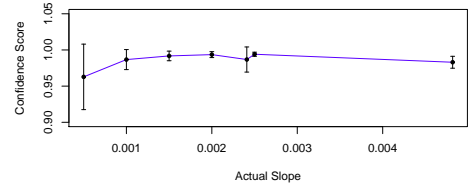
4.2.1 Degradation in Manufacturing Systems

We deploy our profiling system to 8 chemical plants, and collect time series that contain more than 2000 sensors. The sensors are distributed to monitor pressures, temperatures, water flows and sound frequencies of the system. The data has more than 20,000 points, which includes 6 years of monitoring of the system to make sure it captures aging behavior, if any. Since we only focus on long-term behavior, we sample 800 points for each time series and index them using integers from 1 to 800.

After deployment, we run our system on all the collected time series and list the top 9 sensors, showing both strong aging behavior and confidence score, in Table 3. We also plot their original time series and extracted aging curves in Figure 14. From this figure, we can see that the system parts start to have aging behavior at around
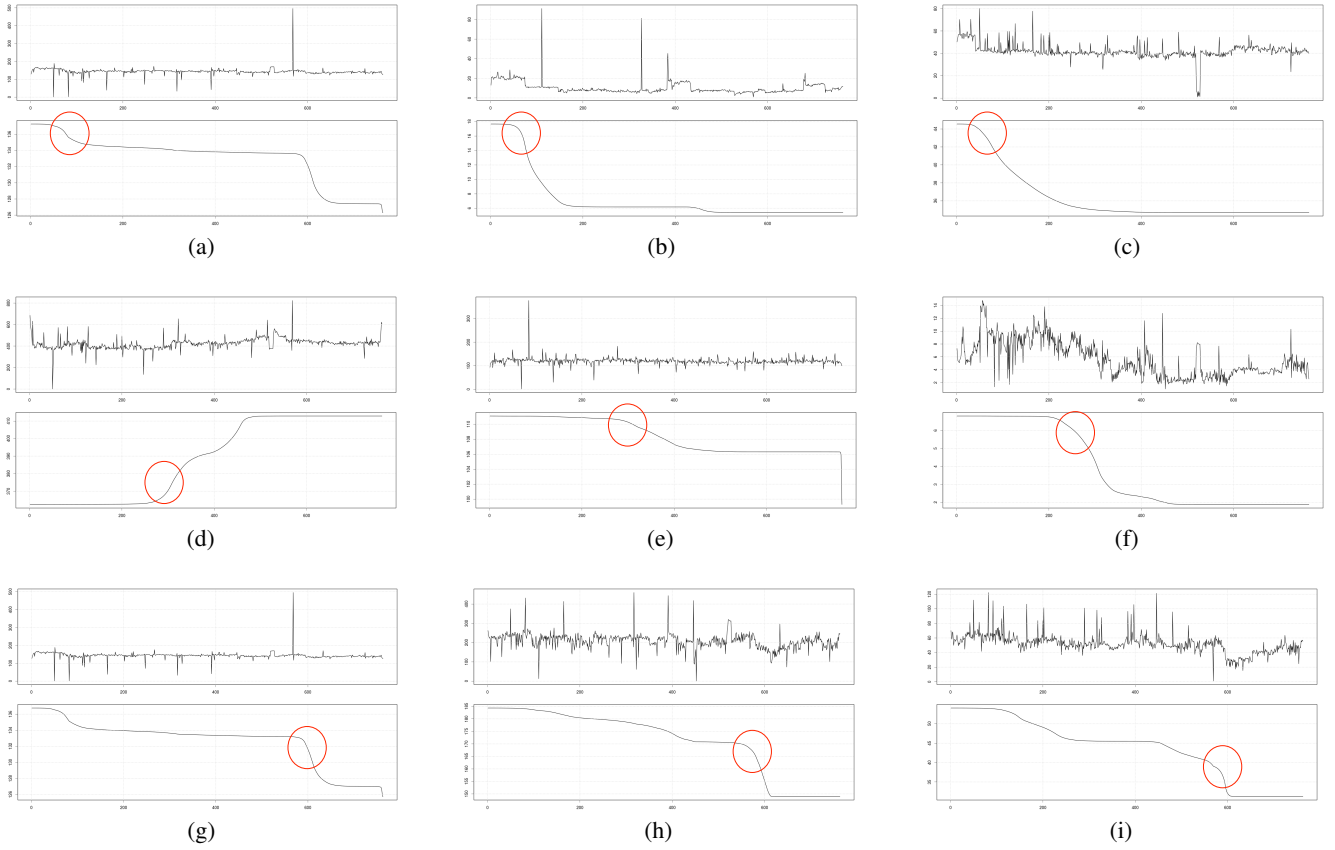
Figure 14: Aging profiling of a chemical plant. (a)-(c), (d)-(f), (g)-(i) show aging behavior start at around 80_{th}, $290th$, 580_{th} points, respectively. Each figure contains the original time series (upper) and the extracted aging curve (lower).

Table 3: Top 9 sensors with highest aging scores

Sensor Name	Aging score	Confidence score
a	0.99	0.99
b	0.99	0.94
c	0.99	0.92
d	1.00	0.96
e	0.99	0.98
f	0.99	0.99
g	0.99	0.99
h	1	0.99
i	0.99	0.99

80_{th}, 290_{th}, 580_{th} points. This is confirmed with domain experts that operate the systems. At around these three points, the pipes of the system that the 9 sensors are monitoring start to have buildups, and the quality of product starts to decrease. This shows our aging profiling engine can accurately profile the aging behavior of the system and observe aging phenomena that cannot be captured by eyes or simple statics.
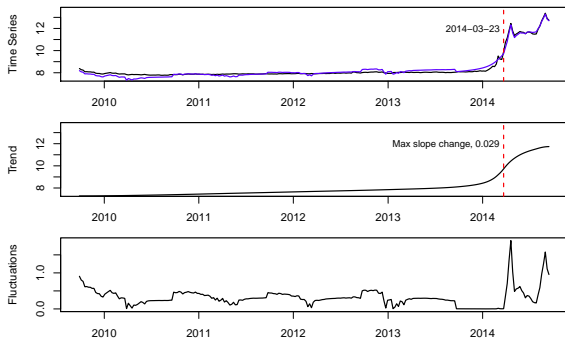
4.2.2   Currency Exchange Rates

Although this paper mostly considers aging trends expressed in physical systems, our techniques are not limited to this domain and can be applied in many scenarios with long term monotonic changes.
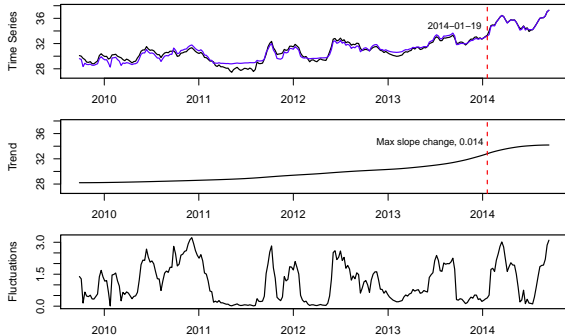
Particularly, we demonstrate how our approach may be used to detect certain events that had influence on the currency exchange rates. Note that we are not claiming applicability of our techniques to the general currency exchange rates monitoring, however, in case of special circumstances our techniques are helpful to detect certain events. Recent events in Ukraine [3] led to a significant decrease of Russian and Ukrainian currencies - Russian Rubles and Ukrainian Hryvnias. Significant changes in the trend appeared in the beginning of 2014 when an unrest started in Ukraine. Figure 15 demonstrates exchange rates between the United States Dollar and aforementioned currencies. Our maximum slope change happens at exactly the same time as that event happens. We obtained currency exchange data from [1] and retrieved the relevant time frame.

4.2.3   Stock Prices

Another application our approach can find is the profiling of long term stock prices. This tracking may be useful to observe changes in a given company success and also to monitor if anything particular happens at some certain time. There are many web resources that provide the information on stock and share prices. For this case study we used data obtained from Yahoo Finance [2] web site and investigated the changes in stock prices of Apple Inc. Figure 16 demonstrates stock prices of Apple Inc. during last 5 years. We can see that the fastest slope change of the trend happened in November 2011 - about the time where iPhone 4S was released, which is detected by our scheme.

(a) Ukrainian Hryvnia

(b) Russian Ruble

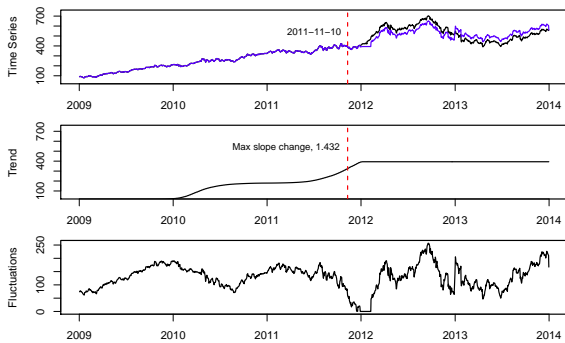Figure 15: Currency Exchange Rates with USD

Figure 16: Apple Stock Prices

5. RELATED WORK

In this section, we discuss related work that has been done in two directions: in time series decomposition and in prediction and analysis of technical system failures.

5.1 Time Series Decomposition Techniques

The time series research, especially trend extraction, has attracted significant efforts in recent decades. In this section, we highlight some relevant techniques and refer an interested reader to a recent survey [4] and therein.

One of the simplest approaches to determine if a time series has a trend is a null hypothesis testing for its presence/absence (H_0 = no change/shift between two time segments) [16]. The other approach is smoothing or filtering [14, 20]. It can be done using different

techniques, including wavelets [12] and the centered moving average that allows getting mean values of a time series in each certain period of time, thus, detecting any systematic change. Another intuitive possibility for trend extraction is fitting a linear or higher level model to the time series data [6]. The common way of fitting a model is a method of least squares [7].

AutoRegressive Integrated Moving Average (ARIMA) is a model-based technique that allows extracting trend and seasonal components from the time series [5]. This approach can be used for both modeling and forecasting, and it is especially useful when there is a certain seasonal component like a yearly change in temperatures or water levels. The drawback of this model is that, in case of absence of domain information about the time series, the user must provide the parameters that may be very subjective, and thus, leading to inadequate results. Therefore, model-based approaches, as it follows from the name, are model sensitive and can be used when there is available information about the model.

One of the parameter free decomposition techniques for time series is Singular Spectrum Analysis (SSA) [10]. SSA allows extraction of alleged trend, seasonal and noise components from time series. The name comes from singular decomposition of a matrix into its spectrum of eigenvalues. The time series can be reconstructed by regrouping different important components. The SSA method is well-known and is used in a broad spectrum of domains [10]. However, it has some drawback that we are aiming to mitigate. We define the trend for aging behavior as a *monotonic* change, whereas, SSA decomposition can return the trend component of arbitrary shape that is not suitable for the problem at hand. In other words, it cannot give clear interpretation of the obtained components, and the trend and high-frequency components are not ensured to be separated.

Another method of time series decomposition is Independent Component Analysis (ICA) [8]. This method allows decomposing of a time series into several additive subparts assuming they are statistically independent and have non-Gaussian distribution. There are two possibilities of how to define the independence of the extracted components. One of them is minimization of mutual information and the other is maximization of negentropy (a measure of distance to normality). This method has the drawback similar to SSA for our problem application: it does not provide necessary condition of trend monotonicity. Moreover, we assume that the signal may change drastically over time, thus, cannot compose the proper ICA set using pieces of different time segments.

5.2 Aging and Degradation Analysis

The problem of aging and performance degradation has attracted significant attention in the research community because this aspect is very important for virtually all technical systems [17, 19]. Here, we discuss only some recent and relevant works.

Djurdjanovic *et al.* [9] proposed a comprehensive framework for analysis and monitoring of the systems based on the time series analysis. They were concerned more with methodology of the collecting and analysis of the data than with the methods of decomposing time series and extracting necessary features. In contrast, in this work we are solving a problem of aging behavior detection that can be expressed as a monotonic change in some properties (particularly, mean value of the time series over time).

The aging behavior also can be expressed in software systems [13]. It was observed that long running software tends to be more failure prone over time, and thus, generally, it is suggested to restart the software systems. However, the definition of a time point when the software must be restarted requires some investigation and research. Authors of [13] demonstrated that the periodical restart

may not work well because the aging mostly depends on the system workload. Thus, it is critical to analyze the aging behavior over time to determine when the system must be restarted.

A similar claim that the scheduled maintenance does not always help avoid the failures and degradation in performance is also presented in [22], in relation to the hardware systems. The work is concerned with log-based predictive analysis in order to monitor the conditions of the operating equipments and provide timely maintenance. The authors built a model in order to classify events and collections of the events as "positive" or "negative" with respect to the failure occurrence. After this, they used this model to predict if the system had come or is coming to a degrading state. The method deals with extracted features and not the shape of the time series.

For data centers failure monitoring, recently a probabilistic method that takes into account network topology and ping data was proposed [15]. This approach is domain specific and initially builds a network model. In our work, as we mentioned above, we are aiming to provide a general approach that could deal with any domain and decompose the time series without usage of a model.

Incident diagnosis from a different prospective was proposed in [18]. Authors presented an approach that allows finding correlation between actual events and time series in order to diagnose incidents. Their approach is able to match events with certain subsequences of time series in order to give a real explanation of the time series shape. However, this work is mostly concerned with finding patterns in time series, whereas our method is dealing with the whole length of time series in order to detect monotonic degradation of an underlying system.

Acknowledgments

6. CONCLUSIONS

In this paper, we profile the long-term degradation in time series by decomposing them into aging and fluctuation components. We formulate the underlying decomposition problem to an optimization problem, and convert it to the Quadratic Programming (QP) formulation whose optimal solution can be obtained. To efficiently solve the formulated problem, we further transform it to a Nonnegative QP problem and propose a method to obtain its solution with iterative updates, which is much faster than using general QP-solvers. We implement and deploy our approach to various physical systems, and evaluate our algorithm with both synthetic and real data. We demonstrate that the proposed approach can approximate the true aging trend with fine resolution (with slope coefficient as small as 10^{-6}), which greatly helps diagnose the corresponding system components.

7. REFERENCES

[1] Currency Exchange Rates. http://www.oanda.com/currency/historical-rates/.

[2] Yahoo Finance. http://finance.yahoo.com/.

[3] 2014 Ukrainian revolution. http://en.wikipedia.org/wiki/2014_Ukrainian_revolution, 2014.

[4] Theodore Alexandrov, Silvia Bianconcini, Estela Bee Dagum, Peter Maass, and Tucker S McElroy. A review of some modern approaches to the problem of trend extraction. *Statistics*, page 3, 2008.

[5] George EP Box, Gwilym M Jenkins, and Gregory C Reinsel. Linear nonstationary models. *Time Series Analysis, Fourth Edition*, pages 93–136, 1976.

[6] K Hung Chan, Jack C Hayya, and J Keith Ord. A note on trend removal methods: the case of polynomial regression versus variate differencing. *Econometrica: Journal of the Econometric Society*, pages 737–744, 1977.

[7] Abraham Charnes, EL Frome, and Po-Lung Yu. The equivalence of generalized least squares and maximum likelihood estimates in the exponential family. *Journal of the American Statistical Association*, 71(353):169–171, 1976.

[8] Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.

[9] Dragan Djurdjanovic, Jay Lee, and Jun Ni. Watchdog agent – an infotronics-based prognostics approach for product performance degradation assessment and prediction. *Advanced Engineering Informatics*, 17(3):109–125, 2003.

[10] James B Elsner and Anastasios A Tsonis. Singular spectrum analysis. *NY: Wiley*, 1996.

[11] Donald Goldfarb and Ashok Idnani. A numerically stable dual method for solving strictly convex quadratic programs. *Mathematical programming*, 27(1):1–33, 1983.

[12] Alexander Grossmann and Jean Morlet. Decomposition of hardy functions into square integrable wavelets of constant shape. *SIAM journal on mathematical analysis*, 15(4):723–736, 1984.

[13] Michael Grottke, Lei Li, Kalyanaraman Vaidyanathan, and Kishor S Trivedi. Analysis of software aging in a web server. *Reliability, IEEE Transactions on*, 55(3):411–420, 2006.

[14] Robert Henderson. Note on graduation by adjusted average. *Transactions of the Actuarial Society of America*, 17:43–48, 1916.

[15] Herodotos Herodotou, Bolin Ding, Shobana Balakrishnan, Geoff Outhred, and Percy Fitter. Scalable near real-time failure localization of data center networks. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1689–1698. ACM, 2014.

[16] Denis Kwiatkowski, Peter CB Phillips, Peter Schmidt, and Yongcheol Shin. Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root? *Journal of econometrics*, 54(1):159–178, 1992.

[17] C Joseph Lu, William Q Meeker, and Luis A Escobar. A comparison of degradation and failure-time analysis methods for estimating a time-to-failure distribution. *Statistica Sinica*, 6(3):531–546, 1996.

[18] Chen Luo, Jian-Guang Lou, Qingwei Lin, Qiang Fu, Rui Ding, Dongmei Zhang, and Zhe Wang. Correlating events with time series for incident diagnosis. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1583–1592. ACM, 2014.

[19] Felix Salfner, Maren Lenk, and Miroslaw Malek. A survey of online failure prediction methods. *ACM Computing Surveys (CSUR)*, 42(3):10, 2010.

[20] Abraham Savitzky and Marcel JE Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, 36(8):1627–1639, 1964.

[21] Fei Sha, Lawrence K Saul, and Daniel D Lee. Multiplicative updates for nonnegative quadratic programming in support vector machines. In *Advances in neural information processing systems*, pages 1041–1048, 2002.

[22] Ruben Sipos, Dmitriy Fradkin, Fabian Moerchen, and Zhuang Wang. Log-based predictive maintenance. In *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1867–1876. ACM, 2014.