Michał GRABOWSKI[1]

O PEWNYCH HEURYSTYKACH DYNAMICZNEGO DOBIERANIA WSPÓŁCZYNNIKÓW WYGŁADZANIA W ALGORYTMACH PROGNOZY

Streszczenie

Prognozy obliczane przez klasyczne algorytmy wygładzania wykładniczego zależą od przyjętych współczynników (wygładzania wartości α i wygładzania trendu β). Przyjmowane wartości tych współczynników są zazwyczaj inne dla różnych analizowanych danych i często zależą od intuicji i doświadczenia osoby eksplorującej dane. W pracy proponujemy opartą o ideę entropii heurystykę automatycznego obliczania współczynnika α przez powiązanie go z entropią błędu prognozy ostatnich n prognoz. Uważamy, że ta entropia może być przyjęta jako miara systematyczności zachowania się błędu prognozy. Zmodyfikowane wersje algorytmów wygładzania wykładniczego zostały wstępnie przetestowane na 120-stu danych. W przypadku różnych modyfikacji algorytmu podwójnego wygładzania dostajemy na danych testowych nieznacznie lepszy błąd średnio-kwadratowy prognozy (rzędu 5%-8%) i nieznacznie lepszą systematyczność zachowania się błędu prognozy. Mierzona proponowanym sposobem systematyczność błędu prognozy jest wyraźnie lepsza niż w różnych wersjach algorytmów średnich ruchomych. Uważamy, że uzyskane wyniki wstępnego eksperymentu pokazują, że idea zasługuje na dalsze poważniejsze eksperymenty, np. na szeregach czasowych próbek czasu podróży pakietu TCP w połączeniu TCP. Prezentowana idea dynamicznego obliczania współczynnika wygładzania pozwala zaproponować pewną modyfikację algorytmu klasycznej specyfikacji TCP obliczania czasu oczekiwania na potwierdzenie (ang. *Retransmission Time Out*).

Abstract

Some of the classic algorithms computing forecasts depend on accepted smoothing factors (value smoothing factor α and trend smoothing factor β). Usually, accepted values of these factors are different for different analyzed data. It happens quite often that they depend on intuition and experience of data explorer. We propose a heuristics of dynamic computation of the factor α based on entropy of some recent forecast's errors. We think that this entropy can be used as a measure of error stability. The modified versions of algorithms have been tested on the 120 element data set. Several modifications of classic algorithms give considerably better mean forecast error (order of magnitude 5%-8%) compared with the classic versions of algorithms. The forecast error stability is clearly better than for moving mean classic algorithms. We propose also the modification of the classic TCP protocol algorithm for computing retransmission time out. The modification is based on the introduced here idea of using forecasts' error entropy.

[1] Dr hab. Michał Grabowski jest profesorem w Warszawskiej Wyższej Szkole Informatyki.

1 WSTĘP

W pracy tej chcemy zaproponować pewną ideę dynamicznego obliczania współczynnika wygładzania wartości α i współczynnika wygładzania trendu β w algorytmach wykładniczego wygładzania obliczania prognozy z szeregu czasowego danych. Współczynniki wygładzania wartości (α) i wygładzania trendu (β) zwykle ustalane są osobno dla każdego zestawu analizowanych danych i często zależą od intuicji i doświadczenia osoby eksplorującej dane.

Proponujemy pewną heurystykę obliczania współczynnika α poprzez powiązanie tego współczynnika z entropią błędu prognozy na n ostatnich danych. Proponujemy też dwie różne idee dynamicznego obliczania współczynnika β. Parametr n pozostaje swobodny i musi być ustalony przed rozpoczęciem analizy danych. Będziemy nazywali go rozmiarem okna entropii. Zagadnienie sterowania rozmiarem okna entropii to osobny problem, którego w tej pracy nie będziemy poruszać. Zdefiniujemy najpierw podstawowe pojęcia szeregu czasowego, okna entropii i entropii błędów prognozy w oknie. Uzasadnimy pogląd, że entropia błędów prognozy okna może służyć jako miara systematyczności zachowania się błędu prognozy w oknie entropii. W następnych rozdziałach zdefiniujemy klasyczne oraz zmodyfikowane wersje algorytmów wygładzania wykładniczego, zmodyfikowaną wersję algorytmu klasycznej specyfikacji protokołu TCP obliczania czasu oczekiwania na potwierdzenie (ang. *Retransmission Time Out*). Omówimy wyniki przeprowadzonego wstępnego eksperymentu i naszkicujemy zaproponowany scenariusz eksperymentu ze zmodyfikowanym algorytmem obliczania czasu oczekiwania na potwierdzenie, który to czas jest jednym z podstawowych narzędzi protokołów warstwy transportowej adaptowania tempa nadawania do sygnałów przeciążenia Intersieci.

Szereg Czasowy Danych Liczbowych

Szereg czasowy to ciąg nieujemnych liczb rzeczywistych $X_{0,}\ X_{1,}..., X_{i,}..., X_{m}$. W punkcie czasowym i wartością danej jest X_i. Na przykład, punkt i to kolejny miesiąc, natomiast X_i to wartość sprzedaży w tym miesiącu.

Okno przesuwne entropii

Dla zadanego $n \in N$ i punktu czasowego t,

$$O\ (t, n) = \{X_{t-n+1},..., X_t\}$$

to okno rozmiaru n w punkcie t (n danych „do tyłu” od punktu t).

Entropia okna

Zakładamy, że dany jest pewien algorytm obliczania kolejnych prognoz

$$F_{k+1,}..., F_{m,}$$

gdzie k jest rozmiarem odcinka początkowego szeregu danych, potrzebnego do wyliczania kolejnych prognoz

$$F_{k+1,}..., F_{m}.$$

dla

$$i = t\text{-}n+1,..., t,\ b_i = |X_i - F_i|,$$

czyli b_i jest błędem prognozy w punkcie czasowym i.

Niech s będzie sumą wszystkich błędów b_i, czyli

$$s = \sum b_i.$$

Znormalizowane błędy definiujemy następująco:

$$p_i = b_i/s,\ 0 \leq p_i \leq 1,\ \sum p_i = 1.$$

Entropię okna rozmiaru n w punkcie t definiujemy następująco:

$$E\ (t, n) = -\sum p_i * \log\ (p_i),$$

gdzie logarytm bierzemy przy podstawie n.

Idea powyższego wzoru jest następująca. Interpretujemy wartości p_i jako prawdopodobieństwo tego, że w i-tej prognozie popełnimy bezwzględny błąd b_i. Wiadomo, że entropia prawdopodobieństw mierzy to, w jakim stopniu wartości b_i mniej więcej z równym prawdopodobieństwem zostaną wylosowane. Jeżeli prawdopodobieństwa p_i są mniej więcej takie same, entropia E (t, n) jest duża, jeżeli wśród prawdopodobieństw p_i niektóre są dominujące, entropia E (t, n) jest mała. Ze względu na związek pomiędzy p_i a b_i ($p_i = b_i/s$), możemy przyjąć, że entropia E (t, n) mierzy to, w jakim stopniu błędy bezwzględne prognoz dla punktów czasowych z okna są mniej więcej takie same, czyli możemy przyjąć, że E (t, n) mierzy systematyczność zachowania się błędu prognozy dla punktów czasowych z okna entropii. Im entropia okna większa, tym większa systematyczność zachowania się błędu prognozy. Innymi słowy, E (t, n) jest pewnym indeksem rozproszenia wartości b_i, $0 \leq E\ (t, n) \leq 1$. Idea ta jest podstawą proponowanych w części drugiej modyfikacji znanych algorytmów wykładniczego wygładzania obliczania prognozy oraz proponowanej w części trzeciej modyfikacji algorytmu klasycznej specyfikacji protokołu TCP obliczania czasu oczekiwania na potwierdzenie (ang. RTO – *Retransmission Time Out*).

2 ALGORYTMY WYGŁADZANIA WYKŁADNICZEGO OBLICZANIA PROGNOZY

Algorytmy wygładzania wykładniczego zmniejszają (w stosunku do np. algorytmów średnich ruchomych) wrażliwość metody na pojedyncze znaczne odchylenia próbek od średniej. Pewną wadą tych algorytmów jest to, że wymagają one na ogół przetworzenia znaczącego odcinka początkowego analizowanego szeregu czasowego by obliczyć dane startowe algorytmu.

2.1 Algorytm pojedynczego wygładzania wykładniczego

Dane:

- k – długość odcinka początkowego danych z których wyliczamy dane startowe
- v – ostatnia wartość tego odcinka, t. j. $v = X_k$
- s – średnia wartości początkowego odcinka szeregu czasowego
- a – współczynnik wygładzania ($0 \le a \le 1$)

Intuicja współczynnika wygładzania:
im α mniejsze, tym większą wagę przykładamy do poprzedniej obliczonej prognozy a mniejszą do poprzedniej (przed obliczaną prognozą) wartości szeregu czasowego;
im α większe, tym większą wagę przykładamy do poprzedniej (przed prognozą) wartości szeregu czasowego a mniejszą do poprzedniej obliczonej prognozy.

Działanie

Wzór na prognozę pojedynczego wygładzania wykładniczego:

$F_{k+1} = a^*v + (1\text{-}a)\ ^*s$ (pierwsza prognoza po odcinku początkowym; $v = X_k$)

$F_i = a^*\ X_{i-1} + (1\text{-}a)\ ^*F_{i-1}$ (dla $i \ge k+2$)

Uwaga: To jak duży jest brany pod uwagę odcinek początkowy i jaki jest przyjęty współczynnik wygładzania na ogół wynika z intuicji i doświadczenia eksploratora danych.

Proponowana modyfikacja algorytmu pojedynczego wygładzania

Wprowadzamy dynamiczną zależność współczynnika wygładzania α od ostatnich n wartości szeregu czasowego. Rozmiar n będzie wartością wejściową algorytmu. Zamienimy zatem dobieranie wartości wejściowej α o skali ciągłej na dobieranie wartości wejściowej n o skali dyskretnej.

Heurystyka

Im większa jest systematyczność błędu prognozy dla ostatnich n wartości tym większy wpływ na kolejną obliczaną prognozę ma ostatnia przed punktem czasowym kolejnej obliczanej prognozy, wartość szeregu czasowego.

Powyższa heurystyka sformalizowana jest jak następuje.

Dane:

- k – długość odcinka początkowego szeregu
- v – ostatnia wartość tego odcinka, t. j. $v = X_k$
- s – średnia wartości początkowego odcinka szeregu czasowego
- n – rozmiar okna entropii

Działanie

Obliczamy pierwsze n prognoz

$$F_{k+1}, F_{k+2}, ..., F_{k+n}$$

zgodnie z klasycznym algorytmem z pewnym przyjętym a. W ten sposób możemy obliczyć entropię E (k+n, n) pierwszego okna entropii

- dla i = k+n+1,..., m
- α (i) = E (i, n) $F_i = \alpha (i\text{-}1)^* X_{i-1} + (1\text{-}\alpha (i\text{-}1)) ^*F_{i-1.}$

Przedstawione w części 2.3 wyniki wstępnego eksperymentu pokazują, że zaproponowana modyfikacja algorytmu ma pewne zalety. Błąd średnio-kwadratowy algorytmu zmodyfikowanego jest nieznacznie gorszy niż metody klasycznej, natomiast systematyczność błędu mierzona średnią entropią okna jest dla algorytmu zmodyfikowanego nieznacznie lepsza. Błąd średnio-kwadratowy algorytmu zmodyfikowanego przy optymalnym rozmiarze okna (n=5) jest praktycznie taki sam jak błąd algorytmu klasycznego przy optymalnym α (α=0,2).

2.2 Algorytm podwójnego wygładzania wykładniczego

Algorytm pojedynczego wygładzania nie inkorporuje dobrze bieżącej tendencji. Podwójne wygładzanie wykładnicze ma inkorporować bieżącą tendencję zmiany danych (albo inaczej: bieżącego trendu). Tak samo jak pojedyncze wygładzanie, algorytm podwójnego wygładzania potrzebuje pewnego znacznego początkowego odcinka szeregu czasowego, by wygenerować startowe wartości do obliczania kolejnych prognoz. Obliczenie algorytmu podwójnego wygładzania zależy od dwóch współczynników: współczynnika wygładzania wartości α i współczynnika β wygładzania trendu.

Dane:

k – liczba wartości początkowego odcinka szeregu czasowego

Z początkowego odcinka wyliczane jest równanie regresji liniowej:

$$y = b + a^*t$$

y – wartość w punkcie t prostej regresji liniowej

t – punkt czasowy,

a – tendencja odcinka początkowego (średni trend odcinka początkowego)

s – średnia wartości odcinka początkowego szeregu czasowego

v – wartość w ostatnim punkcie odcinka początkowego szeregu czasowego ($v = X_k$)

α – współczynnik wygładzania danych

β – współczynnik wygładzania trendu ważonego

Struktury danych (obiekty którymi operuje algorytm)
X_i – rzeczywista wartość danych w punkcie czasowym i szeregu
S_i – wygładzone dane w punkcie czasowym i
TF – czynnik trendu bieżącego w punkcie czasowym i danych wygładzonych
WT_i – wygładzony trend w punkcie czasowym i
F_i – prognoza w punkcie czasowym i

Działanie
Wielkości startowe (przed pierwszą prognozą)

$$F_k = v + a$$

(średnia odcinka początkowego + średni trend odcinka początkowego, k jest ostatnim punktem odcinka początkowego)

$$S_k = \alpha * v + (1-\alpha) * F_k$$
$$TF_k = S_k - v$$
$$WT_k = \beta * TF_k + (1-\beta) * \alpha$$

Kolejne wartości prognozy, wygładzonej danej, czynnika trendu i wygładzonego czynnika trendu:

dla $t>k$:

$$F_t = S_{t-1} + WT_{t-1}$$
$$S_t = \alpha * X_t + (1-\alpha) * S_{t-1}$$
$$TF_t = S_t - S_{t-1}$$
$$WT_t = \beta * TF_t + (1-\beta) * WT_{t-1}$$

Kolejną prognozę F_t obliczamy ze stanu struktur danych w punkcie czasowym t-1. Jeżeli kolejna wartość rzeczywista X_t jest już znana to możemy obliczyć kolejny stan struktur danych w punkcie czasowym t: wartość wygładzoną S_t, czynnik trendu TF_t oraz wygładzony czynnik trendu WT_t.

Proponowana modyfikacja algorytmu podwójnego wygładzania

Tak jak dla algorytmu pojedynczego wygładzania, proponowana modyfikacja polega na wprowadzeniu dynamicznej zależności współczynników α, β wygładzania od n ostatnich danych i n stanów struktur danych. Jako a przyjmiemy entropię okna o przyjętym rozmiarze n, tak jak w algorytmie pojedynczego wygładzania. Motywacja tam prezentowana (α zależy od systematyczności błędu prognozy) tutaj wydaje się być także zasadna. Dla sformalizowania dynamicznej zależności współczynnika β mamy dwie różne idee.

- β zależy od kąta nachylenia prostej regresji poprowadzonej przez punkty z okna entropii,
- β zależy od entropii n ostatnich wartości czynnika trendu TF.

Im większy jest kąt nachylenia prostej regresji, tym większy powinien być wpływ bieżącego czynnika trendu na prognozę, czyli tym większe β. Sterowany tą ideą algorytm obliczania β w punkcie czasowym t jest następujący:

- oblicz prostą regresji względem punktów z okna entropii o przyjętym rozmiarze n,
- β = procent kąta prostego wartości bezwzględnej kąta nachylenia prostej regresji, tzn. β = (wartość bezwzględna kąta nachylenia prostej regresji/kąt prosty).

Policzone β jest w ten sposób w granicach od 0 do 1. Ponieważ kolejne punkty czasowe t zmieniają się o 1, natomiast wartości X_t mogą się zmienić bardzo znacznie (np. w tysiącach), w procesie obliczania prostej regresji musimy zazwyczaj odpowiednio przeskalować wartości X_t, co czynimy w naszym programie obliczającym wyniki klasycznych i zmodyfikowanych wersji algorytmów na wybranych danych testowych.

Aby sformalizować drugą ideę, zdefiniujemy entropię E_{TF} (t, n), n wartości czynnika trendu TF w punkcie czasowym t:

- niech s będzie sumą wartości czynnika trendu $TF_{t-n+1}, \ldots, TF_t$, $s = \sum TF_i$,
- następujące wielkości $p_i = TF_i/s$ sumują się do 1,
- definiujemy $E_{TF}(t, n) = -\sum p_i * \log(p_i)$,

gdzie logarytm bierzemy przy podstawie n.

Proponowana heurystyka jest następująca: im bardziej wśród wartości trendu TF wartości wygładzonych są wartości dominujące, czyli mała entropia trendu E_{TF} (t, n), tym bardziej powinien ważyć trend w obliczeniu kolejnej wygładzonej wartości WT, a więc większe powinno być β liczone w punkcie t. Definiujemy więc algorytm obliczania β w punkcie t następująco.

$$\beta = 1 - E_{TF}(t, n)$$

Zmodyfikowany algorytm podwójnego wygładzania wykładniczego definiujemy jak następuje:

1. oblicz wielkości startowe jak w klasycznej wersji algorytmu;
2. dla przyjętych współczynników α, β oblicz klasycznym algorytmem stany struktur F, S, TF, WT dla punktów czasowych pierwszego po początkowym odcinku k wartości, okna entropii;
3. dla kolejnych punktów czasowych t = k+n+1, k+n+2,... obliczaj kolejne stany struktur F, S, TF, WT zgodnie ze wzorami klasycznej wersji, z tym,

że współczynniki α, β są liczone na nowo dla każdego kolejnego punktu czasowego t.

Nie próbowaliśmy zbudować analitycznego matematycznego modelu zaproponowanych modyfikacji algorytmów wygładzania, aby wywnioskować pewne jakości zachowania się tych modyfikacji algorytmów. Jest to bardzo trudne zadanie, wymagające zapewne głębokiej wiedzy z układów dynamicznych. Próbujemy zbadać zachowanie się proponowanych modyfikacji algorytmów poprzez eksperymenty programistyczne.

2.3 Wyniki wstępnego eksperymentu

Zaproponowane modyfikacje algorytmów wygładzania wykładniczego zostały przetestowane na szeregu czasowym 120-stu danych miesięcznej sprzedaży pewnej firmy. Dane te zostały zaczerpnięte z pracy Roberta Trueblood'a i Johna Lovett'a, *Zastosowanie języka SQL do analizy statystycznej i eksploracji danych* [2], gdzie zostały użyte do ilustracji działania klasycznych wersji algorytmów wygładzania wykładniczego jak i do ilustracji działania algorytmów średnich ruchomych. Poniżej przedstawiono wyniki programistycznego eksperymentu.

Algorytm pojedynczego wygładzania

Błąd średnio-kwadratowy algorytmu zmodyfikowanego jest nieznacznie (rzędu 2% – 5%) gorszy niż metody klasycznej, natomiast systematyczność błędu mierzona średnią (po wszystkich oknach) entropią okna jest dla algorytmu zmodyfikowanego nieznacznie lepsza. Błąd średnio-kwadratowy algorytmu zmodyfikowanego przy optymalnym rozmiarze okna (n=5) jest praktycznie taki sam jak błąd algorytmu klasycznego przy optymalnym a (α=0,2), natomiast systematyczność błędu algorytmu klasycznego dla optymalnego α=0,2 jest gorsza niż systematyczność błędu algorytmu zmodyfikowanego przy optymalnym rozmiarze okna n=5. Sprawdziliśmy przy okazji błąd i systematyczność błędu (jak poprzednio, mierzoną średnią entropią okna) algorytmów średnich ruchomych. Algorytmy te mają (przy odpowiednich wagach) lepszy błąd średnio-kwadratowy niż rozważane wersje algorytmów pojedynczego wygładzania, natomiast, tak jak tego należało oczekiwać (bowiem nie inkorporują trendu), mają zdecydowanie gorszą systematyczność błędu.

Algorytm podwójnego wygładzania

Zaimplementowane zostały obie wersje modyfikacji algorytmu: ta z ideą prostej regresji i ta z ideą entropii okna wartości trendu. Wersja klasyczna algorytmu badana była ze współczynnikiem α=0,3 i dla czterech współczynników β=0,1, 0,2, 0,3, 0,4.

Wersje zmodyfikowane algorytmu były badane z obliczanym α i z obliczanym β w obu wersjach. Algorytm zmodyfikowany został także przetestowany dla obliczanego α i stałych czterech wartości β=0,1, β=0,2, β=0,3, β=0,4. Testy zostały przeprowadzone dla trzech rozmiarów okna entropii: n=4, n=5, n=10. Wyniki są przedstawione w trzech tabelach dla algorytmu zmodyfikowanego z obliczaniem β w wersji poprzez prostą regresji.

Tabela 1

n=4	***alfa=0,3 wersja klasyczna***		***obliczane alfa wersja zmodyfikowana***	
	błąd	***Sr. Entr.***	***błąd***	***Sr. Entr.***
obliczane beta			2226	0,7895
β=0,1	2232	0,7603	2082	0,7181
β=0,2	2334	0,7745	2138	0,7781
β=0,3	2448	0,7821	2212	0,8141
β=0,4	2573	0,7767	2288	0,8227

Tabela 2

n=5	***alfa=0,3 wersja klasyczna***		***obliczane alfa wersja zmodyfikowana***	
	błąd	***Sr. Entr.***	***błąd***	***Sr. Entr.***
obliczane beta			2343	0,7716
β=0,1	2232	0,7585	2089	0,7585
β=0,2	2334	0,7716	2165	0,7797
β=0,3	2448	0,7781	2242	0,7951
β=0,4	2573	0,7737	2332	0,8072

Tabela 3

n=10	***alfa=0,3 wersja klasyczna***		***obliczane alfa wersja zmodyfikowana***	
	błąd	***Sr. Entr.***	***błąd***	***Sr. Entr.***
obliczane beta			2605	0,7319
β=0,1	2232	0,7271	2166	0,7261
β=0,2	2334	0,7381	2246	0,7366
β=0,3	2448	0,7749	2322	0,7501
β=0,4	2573	0,7452	2397	0,7582

Dla wersji obliczania współczynnika β poprzez entropię n ostatnich wartości trendu TF, błędy i entropie okna są następujące:

dla n=4, błąd = 2165, średnia entropia = 0,7592,
dla n=5, błąd = 2161, średnia entropia = 0,7791,
dla n=10 błąd = 2191, średnia entropia = 0,7336.

Wnioski z wyników eksperymentu można wyciągnąć następujące:

- Heurystyka poprzez prostą regresji obliczania β wydaje się być niewiele warta;
- Heurystyka poprzez entropię wartości trendu jest lepsza: daje lepszy błąd średnio-kwadratowy i lepszą systematyczność błędu (dla optymalnego okna n=5) niż metoda klasyczna dla wszystkich, z wyjątkiem jednego, badanych wartości β;
- Błąd metody z obliczanym α i stałym β jest lepszy od błędu metody klasycznej, zarówno jak i systematyczność błędu, dla mniejszych rozmiarów okna entropii; Systematyczność błędu metody z obliczanym α i stałym β rośnie wraz ze wzrostem rozmiaru okna entropii i jest lepsza od systematyczności błędu metody klasycznej;
- Wydaje się, że istnieje zależność: minimalny błąd jest osiągany przy minimalnej entropii okna;
- Systematyczność algorytmów z podwójnym wygładzaniem jest znacznie lepsza niż systematyczność błędu algorytmów z pojedynczym wygładzaniem.

Powyższych wniosków nie można w tej chwili uznać za należycie wiarygodne. Należy przeprowadzić wiele eksperymentów na różnych poważnych zestawach danych i dopiero, jeżeli wyniki takich eksperymentów potwierdziłyby nasze wnioski, ich wiarygodność można by zapewne uznać za zadawalającą. Niestety, nie mamy i chyba nie będziemy mieli relewantnego modelu analitycznego.

3 ALGORYTM KLASYCZNEJ SPECYFIKACJI PROTOKOŁU TCP OBLICZANIA CZASU OCZEKIWANIA NA POTWIERDZENIE I JEGO PROPONOWANA MODYFIKACJA

Jednym z podstawowych zadań protokołu TCP jest osiągnięcie niezawodności przesyłania w połączeniach TCP w zawodnym środowisku Intersieci. Jest to zrealizowane ideą potwierdzeń odbioru pakietów i retransmitowania pakietów, dla których w czasie oczekiwania na potwierdzenie, nie zostało odebrane potwierdzenie odbioru. Protokół TCP nieustannie próbkuje czas podróży pakietów i z próbek czasu podróży na nowo oblicza czas oczekiwania na potwierdzenie (ang. *Retransmission Time Out – RTO*). W najstarszej specyfikacji protokołu TCP algorytm obliczania czasu RTO był oparty o idee algorytmu pojedynczego wygładzania wykładniczego. W kolejnej uznanej za klasyczną specyfikacji zaproponowano algorytm oparty o idee algorytmu podwójnego wygładzania wykładniczego. Nowy algorytm dał wyraźnie lepszą przepływność! Zdefiniujemy teraz algorytm klasycznej specyfikacji protokołu TCP obliczania czasu RTO (Comer [1]).

Różnica = próbka_czasu – stare_RTT
RTT = stare_RTT + δ∗Różnica (= (1 – δ) ∗stare_RTT + δ∗próbka_czasu)
Odchylenie = stare_Odchylenie + ρ* (|Różnica| – stara_Różnica)
Czas_oczekiwania RTO = RTT + η* Odchylenie

Algorytm oblicza RTO z wygładzonej wartości RTT (ang. *Round Trip Time*) czasu podróży pakietu w obie strony. Szereg czasowy, który jest tutaj w tle, to kolejne próbki czasu podróży pakietu w obie strony. Obliczenie algorytmu zależy od przyjętych współczynników wygładzania δ, ρ, η.

Współczynnik wygładzania δ steruje tym, jak bardzo nowa próbka czasu wpływa na prognozowane RTT, a jak wpływa poprzednio obliczone RTT.

Współczynnik wygładzania r steruje wpływem na obliczane odchylenie błędu dwóch ostatnich prognoz czasu RTT, w konsekwencji więc, dwóch ostatnich próbek czasu.

Współczynnik wygładzania h steruje wpływem prognozowanego odchylenia na obliczane nowe RTO.

Współczynniki wygładzania nie są zdefiniowane w specyfikacji. Np. system Unix z Berkeley (BSD 4.3) przyjmuje $\delta = 1/2^3$, $\rho = 1/2^2$. Prawdopodobnie są to sugerowane przez specyfikację TCP wartości.

Wartość bezwzględna różnicy (|Różnica|) we wzorze na nowe Odchylenie oznacza, że nie jest ważne, czy odchylenie próbki na bieżącym etapie jest w górę, czy w dół od bieżącej prognozy RTT. Wygląda na to, że następująca heurystyka jest w tle:

- Jeżeli na poprzednim etapie odchylenie próbki jest w dół (stara_Róznica < 0) to przyjmowane odchylenie jest powiększane (powiększenie moderowane przez ρ);
- Jeżeli na poprzednim etapie odchylenie próbki jest w górę (stara_Różnica > 0) w stopniu większym od wartości bezwzględnej bieżącego odchylenia próbki, to przyjmowane odchylenie jest zmniejszane (zmniejszenie moderowane przez ρ);
- Jeżeli na poprzednim etapie odchylenie próbki jest w górę, ale w stopniu mniejszym od wartości bezwzględnej bieżącego odchylenia próbki, to przyjmowane odchylenie jest zwiększane (powiększenie moderowane przez ρ).

Jest to uwzględnienie szczególnego kontekstu TCP: lepiej przeszacować RTO niż niedoszacować, bowiem każda retransmisja interpretowana jest przez protokół klasycznej wersji TCP jako sygnał przeciążenia, co skutkuje radykalnym (wykładniczym) zmniejszeniem przez protokół tempa nadawania (taki jest efekt działania algorytmu reagowania na przeciążenie protokołu TCP) w połączeniu TCP.

Proponowana modyfikacja algorytmu obliczania RTO

Podobnie jak dla proponowanych modyfikacji algorytmów wygładzania wykładniczego, jako współczynnik wygładzania δ przyjmiemy entropię ostatnich n (n będzie parametrem wejściowym) błędów prognoz czasu RTT w stosunku do rzeczywistych próbek czasu podróży w obie strony pakietu.

Nie mamy na razie idei, w jaki sposób dynamicznie uzależnić od poprzednich wyników obliczeń współczynnik r sterujący wpływem błędu dwóch ostatnich prognoz wartości RTT na oszacowanie nowego odchylenia. Być może, przyjęcie entropii n ostatnich wartości |Różnica|- stara_Różnica jako ρ, byłoby zasadne. Eksperymenty ze zmodyfikowanymi wersjami algorytmu podwójnego wygładzania wykładniczego sugerują, że najistotniejszy jest wpływ (mierzony entropią) systematyczności błędów prognoz w stosunku do wartości rzeczywistych szeregu, na współczynnik wygładzania wartości prognozowanej. Być może pozostawienie pozostałych współczynników wygładzania jako stałych ma sens. Proponowana modyfikacja klasycznego algorytmu obliczania RTO jest więc następująca.

Obliczaj nowe wartości RTO zgodnie ze wzorami algorytmu klasycznego, z tym, że za każdym razem na nowo obliczaj współczynnik δ równy entropii wartości bezwzględnych n ostatnich błędów prognoz czasu RTT. Wartość n jest parametrem wejściowym algorytmu.

Proponowany scenariusz eksperymentu

Najbardziej wiarygodnym byłoby przeprogramować w jądrze Linuxa algorytm obliczania czasu RTO, zrekompilować jądro i przeprowadzić eksperymenty ze sztucznie generowanym ruchem TCP. Jako parametr mierzący efekt modyfikacji algorytmu należałoby przyjąć liczbę retransmisji, a nie czas przekazania komunikatu, bowiem narzut czasowy zmodyfikowanego algorytmu (jest on bardziej skomplikowany) zapewne wydłużyłby czas przekazywania całego komunikatu. To, co naprawdę chcemy zmierzyć, to wpływ modyfikacji na liczbę retransmisji. Istotnie mniejsza liczba retransmisji pakietów algorytmu zmodyfikowanego oznaczałaby, że warto zainwestować bardzo poważny wysiłek w znalezienie takich modyfikacji algorytmu obliczania RTO, które dałyby się zaprogramować z efektywnością porównywalną z efektywnością algorytmu klasycznego.

Realizacji takiego scenariusza mógłby się podjąć tylko bardzo dobry programista systemowy, dodatkowo z doświadczeniem w programowaniu monitorowania ruchu pakietów sieciowych. Trzeba by nawiązać kontakt z jakimś środowiskiem programistów Linuxowych.

Proponujemy inny scenariusz eksperymentu, do przeprowadzenia przez przeciętnie dobrego programistę. Należy wygenerować duży plik z wartościami X_1,..., X_m próbek czasów podróży pakietów TCP w połączeniu TCP, uzyskany z monitorowania sztucznie generowanego ruchu TCP. Następnie programujemy klasyczny i zmodyfikowany algorytm obliczania czasu RTO i testujemy te algorytmy na kolejnych pobieranych z pliku danych, interpretowanych jako kolejne próbki czasu. Jeżeli kolejna próbka czasu przekracza prognozowany czas RTO, zwiększamy liczbę zaobserwowanych retransmisji. Na koniec, porównujemy liczbę zaobserwowanych retransmisji algorytmu klasycznego i zmodyfikowanego.

Pozytywne wyniki eksperymentu wedle scenariusza drugiego byłyby mocnym argumentem za tym, aby przeprowadzić znacznie bardziej pracochłonny eksperyment wedle scenariusza pierwszego.

Na koniec podkreślmy, że istnieje pewna subtelność, która powoduje, że wyniki eksperymentu wedle scenariusza drugiego nie zgadzałyby się w 100 procentach z wynikami eksperymentu wedle scenariusza pierwszego. Subtelnością tą jest definicja w sensie TCP próbki czasu podróży pakietu. Z powodu tzw. problemu niejednoznaczności potwierdzeń, jako próbkę czasu protokół TCP przyjmuje tylko czas podróży pakietu nie retransmitowanego. W przypadku przekroczenia czasu oczekiwania na retransmisję, prognozowany czas RTO jest po prostu dwukrotnie wydłużany i nie zależy od kolejnej próbki czasu (algorytm Karna). Zatem próbka czasu zależy od prognozowanej wartości RTO! Jeżeli algorytm zmodyfikowany lepiej prognozuje w danym momencie RTO, to jego następna próbka czasu może być inna niż następna próbka czasu algorytmu klasycznego. Pozostaje wierzyć, że wpływ wspomnianej subtelności nie zaważyłby znacząco na wiarygodności eksperymentu wedle scenariusza drugiego.

4 ZAKOŃCZENIE

Powtórzmy na zakończenie, że wiarygodna ocena wartości proponowanych modyfikacji rozważanych algorytmów wymaga wielu eksperymentów na poważnych zestawach danych. W tej chwili nie możemy nic wyrokować – wnioski z tylko jednego eksperymentu nie mogą być uznane jako wiarygodne. Możemy tylko powiedzieć, że na razie nie możemy stwierdzić, że proponowane modyfikacje algorytmów są niewiele warte.

Zakładane szeregi czasowe były szeregami danych liczbowych. Proponowane modyfikacje algorytmów mogą być uogólnione na szeregi czasowe danych złożonych

– z pewnej przestrzeni liniowej z określona normą lub metryką. Algorytmy wygładzania wykładniczego dałyby się sensownie zdefiniować w takim kontekście. Moglibyśmy podobnie badać szeregi czasowe rekordów danych liczbowych. Rekordy danych zarówno liczbowych jak i nominalnych wymagałyby rozwinięcia teorii metryk na takich rekordach. Stosowane współcześnie podejście przez włączanie „po Euklidesowemu" do metryki odległości Hamminga kawałków rekordów złożonych z danych nominalnych nie jest, w naszej opinii, adekwatne.

Nie mamy w tej chwili żadnej idei algorytmu sterowania rozmiarem okna entropii. Prawdopodobnie jest to trudne zagadnienie. Wyniki eksperymentu sugerują, że przydałaby się jakaś strategia dochodzenia do optymalnego, względem stosownie zdefiniowanego stosunku błąd – systematyczność błędu, rozmiaru okna entropii.

Literatura

1. Douglas Comer, *Sieci komputerowe TCP/IP, Zasady, protokoły i architektura*, tom 1, WNT 1998
2. Robert Trueblood, John Lovet, *Zastosowanie języka SQL do analizy statystycznej i eksploracji danych*, Wydawnictwo MIKOM 2002