Załącznik 2

Autoreferat: Przetwarzanie sygnału mowy w metodach podnoszących bezpieczeństwo informacyjne

1 Dane osobowe i przebieg zatrudnienia w jednostkach naukowych

1. Imię i nazwisko

   Artur Andrzej JANICKI

2. Posiadane dyplomy, stopnie naukowe – z podaniem nazwy, miejsca i roku ich uzyskania

   - tytuł magistra inżyniera (z wyróżnieniem) w zakresie elektroniki, Wydział Elektroniki i Technik Informacyjnych, Politechnika Warszawska, 1997 r., praca magisterska pt.: „Synteza mowy algorytmem TD-PSOLA".
   - stopień naukowy doktora nauk technicznych (z wyróżnieniem) w dyscyplinie telekomunikacja, Wydział Elektroniki i Technik Informacyjnych, Politechnika Warszawska, 2004 r., rozprawa doktorska pt.: „Selected methods of quality improvement in concatenative speech synthesis for the Polish language".

3. Informacje o dotychczasowym zatrudnieniu w jednostkach naukowych

   - od 2005-02-01 do 2015-02-22 – adiunkt w Zakładzie Systemów Teletransmisyjnych, Instytut Telekomunikacji, Wydział Elektroniki i Technik Informacyjnych, Politechnika Warszawska,
   - od 2015-02-23 – adiunkt w Zakładzie Cyberbezpieczeństwa, Instytut Telekomunikacji, Wydział Elektroniki i Technik Informacyjnych, Politechnika Warszawska.

2 Tytuł osiągnięcia naukowego

Jako osiągnięcie naukowe uzyskane po otrzymaniu stopnia doktora, stanowiące znaczny wkład autora w rozwój dyscypliny naukowej, wskazuję cykl publikacji powiązanych tematycznie zatytułowany **„Przetwarzanie sygnału mowy w metodach podnoszących bezpieczeństwo informacyjne"**. W skład osiągnięcia wchodzą przedstawione w następnym rozdziale publikacje [C1–C9].

3 Wykaz publikacji stanowiących osiągnięcie naukowe

Poniżej zamieszczam listę publikacji wchodzących w skład osiągnięcia naukowego, wraz z krótkim opisem artykułu oraz mojego w nim udziału. Kolejność artykułów odpowiada kolejności ich opisu w dalszej części autoreferatu.

[C1] **Janicki, A.**, Mazurczyk, W., Szczypiorski, K.: Influence of speech codecs selection on transcoding steganography. Telecommunication Systems **59**(3), 305–315 (2014). DOI 10.1007/s11235-014-9937-9. IF=0.705.

Artykuł dotyczy badania efektywności metody steganograficznej TranSteg dla różnych konfiguracji użytych kodeków mowy. Mój wkład polegał na współudziale przy wyborze metodyki badań, zaproponowaniu różnych kombinacji kodeków, zestawieniu środowiska badawczego, przeprowadzeniu większości eksperymentów i współudziale przy formułowaniu wniosków oraz tworzeniu artykułu. Zaproponowałem również nowatorską konfigurację umożliwiającą bezstratne transkodowanie mowy. ***Mój udział szacuję na 45%.***

[C2] **Janicki, A.**, Mazurczyk, W., Szczypiorski, K.: Evaluation of efficiency of transcoding steganography. Journal of Homeland Security and Emergency Management **11**(4), 555–578 (2014). DOI 10.1515/jhsem-2014-0028. IF=0.406.

Artykuł dotyczy wykrywania zastosowania metody steganograficznej opartej na transkodowaniu mowy w sytuacji dostępu do pakietów z podmienionym kodekiem. Mój wkład polegał na zaproponowaniu metody detekcji, wyborze metodyki badań, zestawieniu środowiska badawczego, przeprowadzeniu całości eksperymentów, sformułowaniu większości wniosków oraz utworzeniu większości tekstu. ***Mój udział szacuję na 85%.***

[C3] **Janicki, A.**, Mazurczyk, W., Szczypiorski, K.: Steganalysis of transcoding steganography. Annals of Telecommunications – Annales des Télécommunications **69**(7-8), 449–460 (2014). DOI 10.1007/s12243-013-0385-4. IF=0.699.

Artykuł dotyczy wykrywania zastosowania metody steganograficznej opartej na transkodowaniu mowy w sytuacji dostępu wyłącznie do wyjściowego sygnału mowy. Mój wkład polegał na zaproponowaniu nowatorskiej metody detekcji, opartej na połączeniu użycia mieszanych modeli Gaussa i parametrów kepstralnych, a także na wyborze metodyki badań, zestawieniu środowiska eksperymentalnego, przeprowadzeniu całości eksperymentów, sformułowaniu większości wniosków oraz utworzeniu większości tekstu. ***Mój udział szacuję na 80%.***

[C4] **Janicki, A.**, Mazurczyk, W., Szczypiorski, K.: On the undetectability of transcoding steganography. Security and Communication Networks **8**(18), 3804–3814 (2015). DOI 10.1002/sec.1301. IF=0.720.

Artykuł dotyczy innej metody wykrywania zastosowania metody steganograficznej opartej

na transkodowaniu mowy w sytuacji dostępu wyłącznie do wyjściowego sygnału mowy. Mój wkład polegał na zaproponowaniu metody detekcji opartej na zastosowaniu różnych algorytmów uczących się na podstawie histogramów parametrów kepstralnych, co umożliwiło poprawę detekcji metody TranSteg. Mój wkład to także wybór metodyki badań, zestawienie środowiska badawczego, przeprowadzenie całości eksperymentów, sformułowanie większości wniosków oraz utworzenie większości tekstu. Artykuł zawiera syntezę wyników z wszystkich czterech artykułów dot. steganografii opartej na transkodowaniu mowy. ***Mój udział szacuję na 85%.***

[C5] **Janicki, A.**: Pitch-based steganography for Speex voice codec. Security and Communication Networks (2016). DOI 10.1002/sec.1428. IF=0.720.

W artykule przedstawiłem nowatorską metodę ukrywania informacji w strumieniu telefonii internetowej, polegającą na upraszczaniu przebiegu zmienności parametru opisującego ton krtaniowy, używanego w kodowaniu mowy. Zaprezentowałem jej użycie na przykładzie kodeka Speex. ***Artykuł samodzielny.***

[C6] **Janicki, A.**: SVM-based speaker verification for coded and uncoded speech, pp. 26–30. Proc. 20th European Signal Processing Conference (EUSIPCO). Bucharest, Romania (2012). ISBN 978-1-4673-1068-0.

W artykule przedstawiłem wyniki badań nad weryfikacją mówcy dla oryginalnego sygnału mowy oraz sygnału mowy transmitowanego z użyciem różnych kodeków. Jako metodę weryfikacji użyłem hybrydowego algorytmu SVM-GMM. ***Artykuł samodzielny.***

[C7] **Janicki, A.**: On the impact of non-speech sounds on speaker recognition, *Lecture Notes in Computer Science*, vol. 7499, pp. 566–572. Springer, Berlin, Heidelberg (2012). DOI 10.1007/978-3-642-32790-2_69.

W artykule przedstawiłem oryginalne wyniki badań nad wpływem dźwięków nieartykułowanych na efektywność rozpoznawania mówcy. ***Artykuł samodzielny.***

[C8] **Janicki, A.**, Alegre, F., Evans, N.: An assessment of automatic speaker verification vulnerabilities to replay spoofing attacks. Security and Communication Networks (2016). DOI 10.1002/sec.1499. IF=0.720.

Artykuł dotyczy badania bezpieczeństwa sześciu różnych systemów weryfikacji mówcy poddanych atakowi poprzez odtworzenie nagrania, a także metodom zapobiegania tym atakom. Jedna z metod, oparta na lokalnych wzorcach binarnych, po raz pierwszy została użyta do detekcji tego typu ataków. Mój wkład polegał na współudziale przy wyborze metodyki badań, na zestawieniu środowiska eksperymentalnego (w tym emulacji dziewięciu różnych warunków ataku), przeprowadzeniu znaczej większości eksperymentów i współudziale przy formułowaniu wniosków oraz tworzeniu artykułu. ***Mój udział szacuję na 65%.***

[C9] **Janicki, A.**: Increasing anti-spoofing protection in speaker verification using linear prediction. Multimedia Tools and Applications (2016). DOI 10.1007/s11042-016-3508-x. IF=1.346

W artykule zaprezentowałem nowatorską metodę wykrywania mowy nienaturalnej (tj. syntetycznej lub konwertowanej), opartą na analizie sygnału błędu predykcji liniowej. ***Artykuł samodzielny.***

Niniejszym informuję, że spośród wymienionych wyżej prac jedynie praca [C1] została zgłoszona jako część innego postępowania habilitacyjnego (dr hab. inż. W. Mazurczyk,

udział 45%, stopień przyznany w 2014 r.). Ponadto oświadczam, że żadna z wyżej wymienionych prac nie zawiera wyników badań, przedstawionych w mojej rozprawie doktorskiej, obronionej w 2004 r.

4 Opis osiągnięcia naukowego

Moja praca naukowa, prowadzona w ramach pracy na Politechnice Warszawskiej na stanowisku adiunkta (rozpoczęta jeszcze podczas studiów doktoranckich), dotyczyła różnych aspektów przetwarzania sygnału mowy. Początkowo były to zagadnienia związane z syntezą mowy (czyli generowanie sztucznego sygnału mowy), później także problematyka związana z rozpoznawaniem mowy (cel: rozpoznanie, *co* jest mówione) i rozpoznawaniem mówcy (cel: rozpoznanie, *kto* mówi). Wśród badanych problemów znalazły się również różne aspekty transmisyjne, związane z przesyłaniem mowy w kanałach telekomunikacyjnych, a także zagadnienia związane z ukrywaniem informacji podczas przesyłania sygnału mowy.

Do oceny w procesie habilitacyjnym jako osiągnięcie naukowe chciałbym wskazać cykl publikacji, przedstawiający zagadnienia dotyczące **przetwarzania sygnału mowy w metodach podnoszących bezpieczeństwo informacyjne**. Bezpieczeństwo informacji w usługach telekomunikacyjnych i informatycznych uważam za bardzo ważny problem, dlatego też w ostatnich latach intensywnie zajmowałem się przetwarzaniem sygnału mowy związanym z różnymi aspektami bezpieczeństwa. Badania te prowadzę również obecnie, pracując w Zakładzie Cyberbezpieczeństwa Instytutu Telekomunikacji PW. Dla zwiększenia przejrzystości opisu, cykl publikacji podzieliłem na dwie części:

- Prace związane z ukrywaniem informacji w strumieniu telefonii internetowej (publikacje [C1–C5[1]]).
- Prace związane z bezpieczeństwem systemów rozpoznawania mówcy (publikacje [C6–C9]).

Motywem przewodnim obu wątków osiągnięcia naukowego jest **przetwarzanie sygnału mowy**, w tym wypadku w formie transmisji, kodowania (wraz z kompresją, często kaskadową) oraz rozpoznawania mówcy, połączone z różnymi aspektami **bezpieczeństwa informacyjnego**, takimi jak zachowanie poufności poprzez utajnianie faktu komunikacji (pierwsza część cyklu) lub efektywną weryfikację użytkownika na podstawie głosu (druga część cyklu). Obie części cyklu opisałem w następnych podrozdziałach.

4.1 Prace związane z ukrywaniem informacji w strumieniu telefonii internetowej

Jednym z aspektów bezpieczeństwa, którym się zajmowałem, był temat ukrywania informacji w strumieniu telefonii internetowej. Umieszczanie ukrytych wiadomości podczas przesyłania wiadomości jawnych (tzw. *steganografia*) to technika znana w informatyce od dziesięcioleci. W odróżnieniu od kryptografii, celem steganografii jest nie tyle uniemożliwienie odczytu treści ukrytej wiadomości przez osoby niepożądane, ale zatajenie w ogóle faktu przesyłania dodatkowych informacji. Telefonia internetowa (telefonia IP, *Voice over IP*), ze względu na znaczny

[1] W tym rozdziale odnośniki dotyczą pozycji z cyklu publikacji stanowiących osiągnięcie naukowe, których wykaz zamieszczono w rozdziale 3.

i stale rosnący udział w ruchu przesyłanym w sieciach pakietowych, stała się ostatnio obiektem badań naukowców, szukających nowych sposobów przesyłania ukrytych informacji.

W Instytucie Telekomunikacji PW od 2002 r. pracuje zespół: prof. dr hab. inż. Krzysztof Szczypiorski oraz dr hab. inż. Wojciech Mazurczyk, który zajmuje się różnymi aspektami bezpieczeństwa sieciowego, w tym steganografią. Opracował on różne metody steganograficzne dla telefonii internetowej, które opierają się na modyfikacji pól nagłówków protokołów (np. SIP) lub celowym opóźnianiu pakietów (metoda LACK). W 2012 r. dołączyłem do tej grupy jako specjalista od przetwarzania sygnału mowy.

W odróżnieniu od większości dotychczasowych prac zespołu, metody, w opracowywaniu których brałem wiodący udział, opierały się na modyfikacji pola danych pakietu głosowego, czyli ingerencji w transmitowany sygnał, nie zaś modyfikacji bitów protokołu bądź zmianie zależności czasowych między pakietami. Głównymi celami badań prowadzonych przeze mnie było:

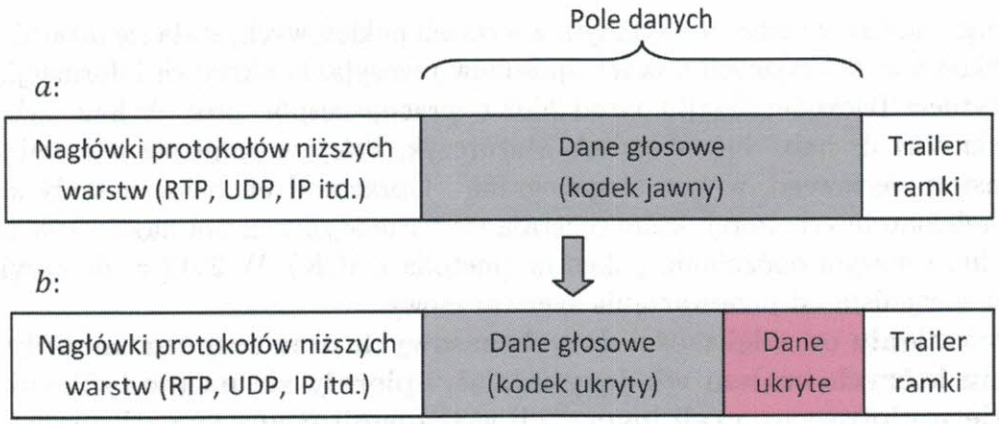
- Opracowanie nowych, efektywnych metod ukrywania informacji w strumieniu telefonii IP, wykorzystujących modyfikację transmitowanego sygnału mowy.
- Sprawdzenie wykrywalności tych metod.
- Opracowanie metod wykrywania przesyłu ukrytej informacji (tzw. metod *steganalizy*) dla metod steganograficznych modyfikujących sygnał mowy.

Poprzez efektywne metody steganograficzne rozumie się metody, które zapewniają:

- Możliwie wysoką *przepływność steganograficzną*, czyli możliwość przesłania dużej ilości ukrytych danych w jednostce czasu.
- Możliwie niski *koszt steganograficzny*, czyli niski spadek jakości transmisji danych jawnych, rozumiany np. jako spadek jakości sygnału mowy lub wzrost opóźnienia transmisji.
- Możliwie niską *wykrywalność* zastosowania algorytmu steganograficznego.

Jedna z metod ukrywania informacji w strumieniu telefonii internetowej, polegająca na modyfikacji pola danych, została zaproponowana przez zespół w roku 2012. Była to metoda polegająca na transkodowaniu, czyli zamianie kodeka użytego do przesłania sygnału mowy lub innego sygnału multimedialnego[2]. Zamiana kodeka następuje z oryginalnego kodeka o większej przepływności, tzw. *kodeka jawnego* (ang. *overt codec*) o większym rozmiarze pola danych pakietu RTP do kodeka o mniejszej przepływności, tzw. *kodeka ukrytego* (ang. *covert codec*), o wyższym stopniu kompresji, mimo że nagłówek protokołu RTP nadal wskazuje poprzedni kodek, czyli kodek jawny. Dzięki zastąpieniu jednego kodeka drugim, w polu danych pakietu głosowego zostaje zaoszczędzone miejsce, które można wypełnić ukrytą informacją (tzw. *steganogramem*, patrz rysunek 2.1). Operacja zamiany kodeków musi być przeprowadzona w taki sposób, by zapewnić możliwie najmniejsze pogorszenie jakości. Opracowana metoda otrzymała nazwę *TranSteg*. We wspomnianej pracy autorzy przedstawili opis weryfikacji tej koncepcji (*proof-of-concept*) z użyciem pary kodeków (kodek jawny/kodek ukryty): G.711/G.726.

[2]Mazurczyk, W., Szaga, P., Szczypiorski, K. Using transcoding for hidden communication in IP telephony. Multimedia Tools and Applications, June 2014, Volume 70, Issue 3, pp 2139-2165.



Rysunek 2.1: Idea działania algorytmu TranSteg: a) pakiet danych w czasie normalnej transmisji głosu, b) pakiet danych w sytuacji, gdy przesyłane są ukryte dane za pomocą metody TranSteg.

Rozwój steganografii z użyciem transkodowania sygnału mowy

Po dołączeniu do zespołu zajmującego się steganografią zająłem się rozwojem metody TranSteg. Prowadziłem prace w celu znalezienia najlepszych par kodeków jawnych i ukrytych z punktu widzenia wprowadzanego kosztu steganograficznego oraz przepływności steganograficznej. Do eksperymentów postanowiłem wykorzystać dziewięć różnych kodeków, stosowanych zarówno w telefonii internetowej, jak i w telefonii stacjonarnej lub mobilnej, takich jak G.711, iLBC, G.723.1, G.726, G.729, GSM 06.10, AMR czy kodek Speex. Jeśli chodzi o kodek Speex, zaproponowałem użycie trzech różnych trybów jego pracy: trybu 7, o najlepszej jakości, trybu 4, o umiarkowanej jakości oraz trybu 2, który jest najniższym trybem dopuszczonym przez twórców kodeka Speex dla sygnału mowy. Dodatkowo, oprócz wymienionych kodeków stratnych, zaproponowałem użycie bezstratnego kodeka G.711.0, co w zamierzeniu miało umożliwić funkcjonowanie metody TranSteg z zerowym kosztem steganograficznym. W sumie do eksperymentów użyłem 12 różnych kodeków, z czego sześć, najczęściej używanych w telefonii IP, było używanych jako kodeki jawne, a 11 jako kodeki ukryte.

W artykule **[C1]** opisałem szczegółowe eksperymenty, które przeprowadziłem dla różnych par kodeków, mierząc oferowaną przepływność steganograficzną oraz koszt steganograficzny, rozumiany tu jako pogorszenie jakości sygnału mowy względem transmisji wyłącznie przy pomocy kodeka jawnego. Do pomiaru jakości użyłem algorytmu PESQ (Perceptual Evaluation of Speech Quality)[3]. Eksperymenty przeprowadziłem w środowisku MATLAB, z zastosowaniem emulacji poszczególnych kodeków. Jako materiał dźwiękowy użyłem 20 jednominutowych nagrań, imitujących sygnał przesyłany w jednym kanale połączenia telefonicznego o średniej aktywności głosowej wynoszącej 46,5%. Nagrania pochodziły z korpusów TSPspeech oraz CORPORA, odpowiednio dla języka angielskiego i polskiego. Nagrania były zrównoważone pod kątem płci mówców.

Uzyskane wyniki pozwoliły na wyznaczenie dziesięciu najbardziej obiecujących par kodeków. W zależności od osiągniętej efektywności różne konfiguracje kodeków podzieliłem na trzy klasy:

- *Klasa 0* – konfiguracje powodujące znikomy spadek lub brak spadku jakości (koszt

[3]Recommendation P.862: Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs. ITU, 2001.

steganograficzny poniżej 0,1 w skali MOS[4];

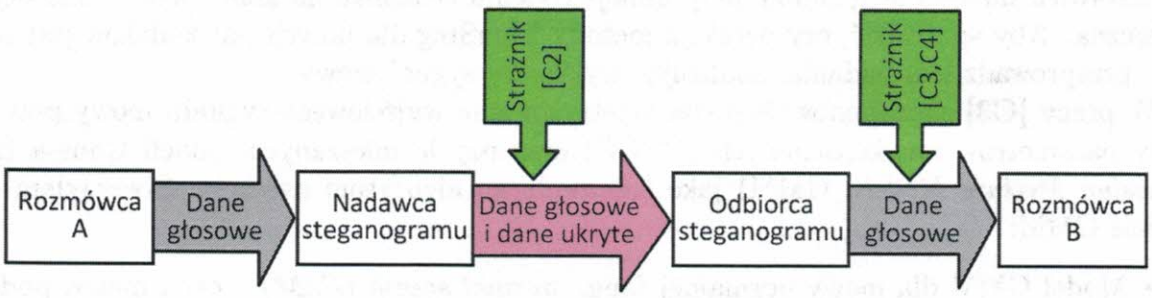
- *Klasa 1* – konfiguracje powodujące mały spadek jakości (koszt steganograficzny pomiędzy 0,1 a 0,5 MOS);
- *Klasa 2* – konfiguracje powodujące umiarkowany spadek jakości (koszt steganograficzny pomiędzy 0,5 a 1,0 MOS).

Okazało się, że najlepszą parę tworzy kodek G.711 jako kodek jawny, szeroko stosowany w telefonii IP ze względu na swą prostotę, wraz z bezstratnym algorytmem G.711.0 (jako kodek ukryty). Taki zestaw kodeków zapewnia wysoką przepływność steganograficzną (ok. 30 kb/s), przy jednoczesnym braku wprowadzanego kosztu steganograficznego tj. przy zachowaniu tej samej jakości rozmowy przed jak i po transkodowaniu. Ta para kodeków znalazła się w klasie 0. Do klasy 1 zaliczyłem następujące pary: Speex(7)/AMR, G.711/Speex(7), iLBC/AMR oraz G.711/AMR. Pięć innych par zaliczyłem do klasy 2. Najbardziej elastycznym algorytmem kodowania mowy do zastosowania jako kodek ukryty okazał się kodek AMR pracujący w trybie 12,2 kb/s – zapewniał on małe spadki jakości, pracując z różnymi kodekami jawnymi, mimo że umożliwiał wydajną kompresję, a przez to efektywną tranmsisję w kanale niejawnym.

Badanie wykrywalności steganografii opartej na transkodowaniu sygnału mowy

W kolejnych pracach [C2, C3, C4] zajmowałem się zagadnieniem wykrywalności ukrytej transmisji z wykorzystaniem metody TranSteg. W poszczególnych publikacjach badałem możliwości steganalizy dla różnych możliwych przypadków położenia strażnika (ang. *warden*), który analizuje przesyłane pakiety:

- Strażnik znajduje się „w środku" kanału telekomunikacyjnego, czyli ma dostęp do pakietu, którego część zajmuje steganogram, a część jest zakodowana innym kodekiem, niż zadeklarowano w nagłówku RTP (temat omawiany w publikacji [C2]).
- Strażnik znajduje się na końcu kanału telekomunikacyjnego, czyli ma dostęp tylko do pakietu, w którym steganogramu już nie ma (został odczytany, „zdjęty" we wcześniejszych węzłach sieci), zaś zawartość głosowa została na nowo zakodowana kodekiem jawnym (temat omawiany w publikacjach [C3, C4]).



Rysunek 2.2: Możliwe warianty lokalizacji strażnika, omawiane w publikacjach [C2–C4].

Pierwszy przypadek, w którym strażnik ma dostęp do struktury bitowej pakietów VoIP w czasie, gdy zawierają one steganogram (czyli gdy kodek jest podmieniony), przedstawiłem w pracy **[C2]**. Opisałem użycie „lekkiego" algorytmu steganalizy, który będzie analizował

[4]ang. *Mean Opinion Score* – uśredniona ocena słuchaczy, wyrażana w skali od 1 do 5

pierwszy bajt pola danych, aby przy niskim nakładzie obliczeniowym sprawdzać, czy użyty kodek odpowiada zadeklarowanej wartości kodeka w polu PT nagłówka RTP. W badaniu użyłem tych kodeków jawnych i ukrytych, które w badaniach opisanych w [C1] uznałem za odpowiednie dla metody TranSteg ze względu na przepływność i koszt steganograficzny. Zastosowałem podejście oparte na danych (*data-driven approach*). Zaproponowałem użycie algorytmu uczącego się (w tym wypadku: drzewa decyzyjnego), który wyszkoliłem, używając zbioru uczącego zawierającego ok. 450 tys. pakietów, zaś eksperymenty z detekcją prowadziłem na zbiorze testowym o podobny rozmiarze, rozłącznym z pierwszym. Analizowałem efektywność detekcji poszczególnych kodeków, mierząc odzew i precyzję ich detekcji. Przedstawiłem także macierz pomyłek, by znaleźć pary kodeków, które łatwo ze sobą pomylić.

Wynikiem badań był wniosek, że przy użyciu analizy pierwszego bajtu pola danych kodek Speex we wszystkich trybach pracy jest łatwy do wykrycia. Przyczyną tego są charakterystyczne bity, znajdujące się w pierwszym bajcie pola danych, w których zakodowany jest tryb pracy kodeka. Warto jednak dodać, że dyskutując ten wynik w publikacji [C2], opisałem sposób obniżenia wykrywalności kodeka Speex, polegający na modyfikacji bądź usunięciu charakterystycznych bitów nagłówka kodeka Speex. Wyniki prac wskazały również pary kodeków, które były łatwo ze sobą mylone – chodzi o pary: G.711/GSM06.10 oraz iLBC/G.723.1. W związku z tym metoda TranSteg, która używałaby tych par kodeków, byłaby trudna do wykrycia. Końcowym wynikiem prac, opisanych w artykule [C2], było uzupełnienie rekomendacji par kodeków, wskazanych w publikacji [C1], o informację dotyczącą stopnia ich wykrywalności, w sytuacji gdy strażnik ma dostęp do pakietów z podmienionym kodekiem. Wspomniane dwie pary kodeków, w których kodek jawny i ukryty są łatwo ze sobą mylone, przydzieliłem do par o wysokiej niewykrywalności. Innym konfiguracjom przypisałem średnią (oba kodeki trudno rozpoznawane) lub niską (oba kodeki łatwo rozpoznawane) niewykrywalność. Oddzielnie sklasyfikowałem przypadki, gdy tylko jawny lub ukryty kodek są łatwo wykrywalne, wskazując dodatkowo, że odpowiednia korekta nagłówka pola danych (np. skopiowanie go z innego kodeka), może znacząco podnieść niewykrywalność.

Kolejne badania, których wyniki opisałem w artykułach [C3, C4], dotyczyły steganalizy metody TranSteg w sytuacji trudniejszej, gdy strażnik znajduje się na końcu kanału telekomunikacyjnego, tam gdzie steganogramu już nie ma. W tej sytuacji kodek zadeklarowany w nagłówku RTP jest (ponownie) zgodny z zawartością pola danych, tak więc prosta steganaliza, taka jak opisana w [C2], nie będzie skuteczna. Jeżeli jako kodek ukryty zostanie użyty kodek bezstratny G.711.0, oryginalny sygnał mowy zostanie bezbłędnie przywrócony i jakakolwiek metoda steganalizy przy umiejscowieniu strażnika na końcu kanału nie będzie skuteczna. Aby sprawdzić, czy detekcja metody TranSteg dla innych par kodeków jest możliwa, przeprowadziłem badania, analizując wyjściowy sygnał mowy.

W pracy **[C3]** zaproponowałem sparametryzowanie wyjściowego sygnału mowy przy pomocy parametrów mel-kepstralnych (MFCC) oraz użycie mieszanych modeli Gaussa (ang. *Gaussian Mixture Models*, GMM) jako trenowalnego algorytmu detekcji. Stworzyłem dwa modele GMM:

- Model GMM dla mowy normalnej (ang. *normal speech GMM*) – czyli mowy, podczas transmisji której nie zaszło transkodowanie, oraz:
- Model GMM dla mowy atypowej (ang. *abnormal speech GMM*) – czyli mowy, podczas transmisji której doszło do zamiany kodeków, czyli mógł zostać użyty algorytm TranSteg.

Oba modele wyszkoliłem na nagraniach 200 mówców pochodzących z bazy TIMIT, odpowiednio z użyciem mowy „czystej" oraz transkodowanej. Eksperymenty z detekcją prowadziłem na

pięciu różnych zbiorach nagrań, rozłącznych ze zbiorem uczącym, pochodzących z korpusów: TSPspeech, TIMIT, CHAINS, CORPORA oraz AHUMADA – był więc to zbiór poszerzony względem eksperymentów opisanych w artykułach [C1, C2]; dodatkowo baza AHUMADA zawierała nagrania w języku hiszpańskim.

Podczas eksperymentów okazało się, że dla różnych par kodeków podczas transkodowania parametry MFCC ulegają w różny sposób modyfikacji, co można spróbować wykorzystać do detekcji użycia metody TranSteg. Przeprowadziłem więc próby z detekcją dla różnych par kodeków, badając dokładność (ang. *accuracy*) rozpoznawania mowy normalnej i atypowej. Dokładność powyżej 80% uznawałem za wysoką, dokładność poniżej 70% uznawałem za niską; dokładność detekcji bliska 50% oznaczałaby wynik zupełnie losowy i całkowitą nieprzydatność detektora. Badałem również dokładność rozpoznawania w funkcji długości sygnału mowy dostępnego do analizy, a także zależność skuteczności rozpoznawania od języka, w którym dokonano nagrań.

Wyniki badań pokazały, że zaproponowana metoda steganalizy umożliwia stosunkowo łatwą detekcję użycia metody TranSteg, jeżeli użyto w niej następujące pary kodeków: G.711/G.726, G.711/G.726, Speex7/G.729, Speex7/iLBC. Eksperymenty dowiodły, że wystarcza ok. 7 s sygnału mowy, by osiągnąć wysoką dokładność rozpoznawania tych par. Z kolei niektóre pary kodeków są bardzo odporne na detekcję przy pomocy zaproponowanej metodą steganalizy – takimi parami są np. kodeki G.711/Speex7, iLBC/AMR i pozostałe konfiguracje, w których kodek AMR występuje jako kodek ukryty. Wyniki dla nagrań w języku innym niż angielski (czyli dla języka polskiego i hiszpańskiego) zwykle były gorsze, co daje się wytłumaczyć tym, że w zbiorze uczącym użyto tylko nagrań w języku angielskim. Co ciekawe, różnice między językami nie rozkładały się równomiernie dla wszystkich kodeków: największe były dla kodeka iLBC użytego jako kodek ukryty, podczas gdy dla kodeków G.726, Speex7 i G.723.1 były pomijalne.

Zgodnie z oczekiwaniami dało się zaobserwować, że istnieje pewna korelacja pomiędzy kosztem steganograficznym a dokładnością detekcji danej pary kodeków – im większy koszt steganograficzny (czyli większy spadek jakości), tym zwykle łatwiej taka para była detekowana. Zaobserwowałem jednak, że nie jest to regułą. Istnieją bowiem pary (np. G.711/G.726, G.711/Speex7), które oferują zbliżony koszt steganograficzny (w tym wypadku: 0,4 MOS), różnią się jednak znacznie pod kątem wykrywalności: pierwsza para jest łatwo wykrywalna przy pomocy zaproponowanej metody (dokładność detekcji ok. 95%), podczas gdy wykrywalność drugiej pary nieznacznie przekracza 60%, czyli jest bardzo niska.

W artykule **[C4]** zaproponowałem jeszcze jedną metodę steganalizy, która mogłaby zostać użyta w sytuacji, gdy strażnik jest umiejscowiony na końcu kanału telekomunikacyjnego. Podobnie jak poprzednio, wykorzystałem parametryzację wyjściowego sygnału mowy przy pomocy parametrów MFCC, ale tym razem postanowiłem analizować ich histogramy. Z każdego nagrania co 10 ms ekstrahowałem 19 parametrów MFCC, używając okna o długości 30 ms. Dla pozyskanych w ten sposób wartości MFCC wyznaczałem 19-wymiarowe znormalizowane histogramy, z użyciem 20 przedziałów dla każdego wymiaru. Następnie wartości 20 częstości dla każdego z 19 wymiarów łączyłem w superwektor o długości 19 x 20 = 380 wartości. Takie superwektory poddawałem dalszej analizie.

Do analizy pozyskanych danych użyłem różnych algorytmów uczących się, takich jak sieci Bayesa, drzewa decyzyjne, algorytm C4.5, maszyna wektorów nośnych (SVM), sztuczna sieć neuronowa (w tym wypadku: perceptron wielowarstwowy, MLP) oraz algorytm AdaBoost. Dodatkowo stosowałem selekcję parametrów wejściowych z użyciem postępującej selekcji krokowej (ang. *Sequential Forward Selection*, SFS). Do uczenia i testowania detekcji użyłem tych samych zbiorów nagrań co we wcześniejszym badaniu [C3]. W wyniku przeprowadzonych eks-

perymentów okazało się, że w części przypadków (np. dla par G.711/iLBC, Speex7/G.729) zastosowane algorytmy przyniosły podobne rezultaty do metody opartej na modelach GMM, zaproponowanej w [C3]. Jednak w niektórych przypadkach – dokładniej, dla większości par używających kodeka iLBC jako kodeka jawnego, wykrywalność użycia transkodowania wyraźnie wzrosła. Algorytmem, który wykrywał te przypadki z największą dokładnością (ok. 85%), okazał się być algorytm oparty na sieciach Bayesa.

Oprócz wyników dla nowej metody steganalizy, artykuł [C4] zawierał podsumowanie ustaleń z publikacji [C1–C4], w tym zaktualizowane rekomendacje dotyczące par kodeków zapewniających niską wykrywalność przy jednocześnie wysokiej przepływności steganograficznej. Zgodnie z tymi ustaleniami najniższą wykrywalność uzyskano dla następujących par: G.711/Speex7, iLBC/GSM06.10 oraz G.711/G.711.0. W artykule powtórzono jednak zastrzeżenie, że podczas używania kodeków Speex oraz G.711.0 należałoby zmodyfikować (lub usunąć) początkowe bajty pola danych – w przeciwnym razie strażnik umiejscowiony w środku kanału telekomunikacyjnego, nawet używając „lekkiej" analizy (np. takiej jak opisana w [C2]), mógłby bez dużego wysiłku wykryć podmianę kodeków.

Informuję dodatkowo, że Rada Wydziału Elektroniki i Technik Informacyjnych PW na posiedzeniu w dniu 23.03.2016 pozytywnie zaopiniowała wniosek o nagrodę zespołową I stopnia Rektora PW za osiągnięcia naukowe – prace badawcze z zakresu steganografii sieciowej w sieciach komunikacyjnych w latach 2014–2015 (w skład zespołu, oprócz mnie, wchodzą K. Szczypiorski, W. Mazurczyk oraz E. Rzeszutko).

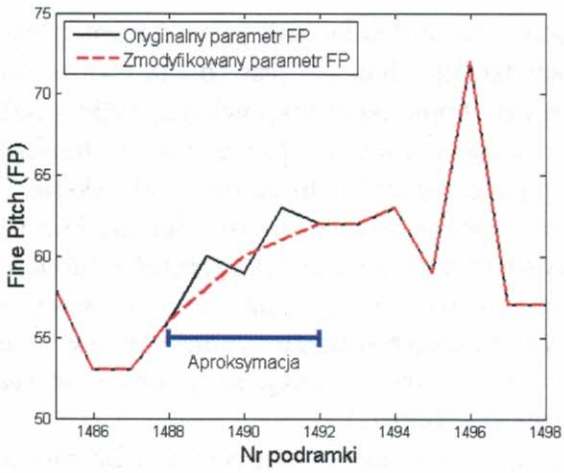
Ukrywanie informacji w parametrze opisującym ton krtaniowy

Oprócz rozwijania metod steganografii i steganalizy związanej z transkodowaniem mowy, prowadziłem prace nad innymi metodami ukrywania informacji w strumieniu telefonii IP, wykorzystującymi modyfikację pola danych. W 2015 r. zaproponowałem **nowatorską metodę ukrywania informacji**, która wykorzystywała jeden z parametrów używanych do zakodowania sygnału mowy – parametr opisujący zmienność tonu krtaniowego[5]. Metodę tę przetestowałem na koderze Speex, który informację związaną z tonem krtaniowym, czyli tzw. częstotliwość podstawową lub częstotliwość $F0$ (ang. *pitch*) przesyła w parametrze nazwanym FP (ang. *Fine Pitch*). Parametr ten ma na celu przesłanie informacji o tym, z jaką częstotliwością wibrują struny głosowe mówcy w danym momencie, co jest wykorzystywane w celu możliwie jak najwierniejszej resyntezy sygnału mowy w dekoderze po stronie odbiorczej.

Zaobserwowałem, że parametr FP dla niektórych fragmentów sygnału mowy zachowuje się w znacznym stopniu deterministycznie - jego wartość albo monotonicznie rośnie, albo maleje. Postanowiłem ten fakt wykorzystać i wykrywać takie fragmenty, po to, aby monotoniczne odcinki parametru FP zastępować ich liniową aproksymacją (patrz rysunek 2.3), a zaoszczędzone bity wykorzystać do stworzenia ukrytego kanału komunikacyjnego. Metodę tę nazwałem *HideF0*, a jej szczegółowy opis wraz z wynikami eksperymentów dla kodeka Speex zamieściłem w artykule **[C5]**, wchodzącym w skład osiągnięcia naukowego.

Algorytm umieszczania ukrytej informacji, przedstawiony w [C5], działa następująco: podczas kodowania mowy sygnał dzielony jest na ramki o czasie trwania 20 ms. W każdej ramce sprawdzany jest przebieg parametru FP by sprawdzić, czy da się go przybliżyć funkcją liniową z błędem mniejszym niż założony próg θ. Jeżeli tak, to w jednym z pól nagłówków niższych warstw modelu OSI ustawiana jest flaga informująca o tym, że dana ramka będzie wykorzysty-

[5] Janicki, A. Novel Method of Hiding Information in IP Telephony Using Pitch Approximation, In: Proc. International Workshop on Cyber Crime (IWCC 2015), Toulouse, France, 2015



Rysunek 2.3: Idea aproksymacji wartości tonu krtaniowego w algorytmie HideF0.

wać aproksymację parametru FP, a następnie trzy spośród czterech wartości parametru FP mogą zostać zastąpione przez ukryte dane. Podczas dekodowania, gdy odbiornik stwierdzi obecność flagi aproksymacji, wówczas ukryte dane są odczytywane, a brakujące wartości parametru FP są przybliżane liniowo na podstawie wartości przesłanych w niezmienionym stanie. Jeśli nie dojdzie do odczytania steganogramu i rekonstrukcji brakujących wartości FP (np. gdy strumień zostanie odebrany przez dekoder nie wspierający algorytmu HideF0), wówczas sygnał mowy nadal jest zrozumiały, choć głos staje się mniej czysty a bardziej „chropawy". Gdy rekonstrukcja nastąpi, wówczas spadek jakości jest zwykle niewielki, a jego wartość zależy od ustawionego progu θ.

W artykule [C5] zamieściłem eksperymenty z wykorzystaniem algorytmu HideF0 dla kodeka Speex dla różnych trybów jego pracy, a także dla różnych wartości progu θ. Przeprowadziłem testy na zbiorze nagrań, pochodzących z bazy TIMIT, zawierającym nagrania 24 mówców męskich i 24 żeńskich, po ok. 30 s sygnału na każdego mówcę. Mierzyłem przepływność steganograficzną oferowaną w różnych konfiguracjach pracy algorytmu, a przy pomocy algorytmu PESQ mierzyłem także spadek jakości mowy, który stanowił koszt steganograficzny. Wyniki badań pokazały, że zaproponowana metoda umożliwia utworzenie bezstratnego algorytmu steganograficznego oferującego ukryty kanał o przepływności ok. 50 b/s, a także kanałów o przepływności 170–200 b/s (zależnie od trybu pracy kodeka Speex) przy niewielkim koszcie steganograficznym, wynoszącym w granicach 0,3 – 0,7 MOS. Inne metody, wykorzystujące również częstotliwość podstawową, oparte na technice najmniej znaczącego bitu (LSB), oferowały znacznie gorsze parametry. Warto dodać, że mimo że zaproponowaną metodą testowałem na kodeku Speex, to z powodzeniem można ją zastosować także w innych kodekach, które przesyłają w strumieniu bitów parametry opisujące ton krtaniowy.

4.2 Prace związane z bezpieczeństwem systemów rozpoznawania mówcy

Równolegle z pracami dotyczącymi przesyłania ukrytych informacji w sygnale mowy prowadziłem badania dotyczące innego obszaru przetwarzania mowy związanego z bezpieczeństwem: systemów rozpoznawania mówcy. Systemy rozpoznawania mówcy to systemy pozwalające ustalić tożsamość użytkownika na podstawie analizy jego głosu. Metody tego rodzaju stosuje się np. w celu weryfikacji użytkownika podczas rozmowy z operatorem infolinii czy banku. Tego rodzaju systemy określa się mianem systemów weryfikacji mówcy (ang. *automatic spe-*

aker verification, ASV).

Systemy weryfikacji mówcy mogą działać zależnie od tekstu, wtedy gdy oczekują od użytkownika konkretnej wypowiedzi, np. hasła – jest to tak zwana *biometria głosowa aktywna*. Jeżeli system ASV nie oczekuje konkretnej wypowiedzi, tylko analizuje dowolną wypowiedź (np. działa w tle podczas rozmowy klienta z pracownikiem banku), wówczas mamy do czynienia z *biometrią głosową pasywną*, czyli niezależną od tekstu. Mimo że opisane poniżej eksperymenty były prowadzone w warunkach braku zależności od tekstu, większość wniosków z nich płynących można wykorzystać również dla systemów biometrii aktywnej.

Gdy do systemu ASV jest dodawany nowy użytkownik, wówczas nagrywa się jego wypowiedź i system na podstawie zarejestrowanego sygnału mowy tworzy dla niego odpowiedni model matematyczny, tzw. model mówcy. Dzieje się podczas procesu określanego jako wdrażanie użytkownika (ang. *user enrollment*).

Aby systemy weryfikacji mówcy działały efektywnie i bezpiecznie, powinny spełniać następujące warunki:

- powinny zapewniać niskie prawdopodobieństwo fałszywej akceptacji (ang. *false acceptance rate*, FAR), to jest akceptacji użytkownika nieuprawnionego, oraz
- powinny zapewniać niskie prawdopodobieństwo błędnego odrzucenia (ang. *false rejection rate*, FRR), to jest odrzucenia użytkownika uprawnionego.

Najczęściej w badaniach systemów ASV podaje się wartość błędu zrównoważonego (ang. *equal error rate*, EER), czyli wartość, dla której oba powyższe prawdopodobieństwa są sobie równe. Należy podkreślić, że błędy fałszywej akceptacji mogą być powodowane dwiema przyczynami: niedoskonałością pracy algorytmu rozpoznawania mówcy albo/oraz celowymi atakami na system ASV. Opisane niżej prace dotyczą różnych zagadnień związanych z badaniem systemów ASV, w tym tematów związanych z ich bezpieczeństwem.

Badania efektywności działania systemów rozpoznawania mówcy dla mowy o różnej jakości

W artykule **[C6]** opisałem eksperymenty z weryfikacją mówcy z uwzględnieniem zniekształceń sygnału podczas transmisji w telefonii stacjonarnej, mobilnej i pakietowej. Jest to ważny problem, gdyż weryfikacja użytkownika z wykorzystaniem głosu często odbywa się zdalnie, z użyciem usług telekomunikacyjnych: np. klient może się z kontaktować z bankiem poprzez publiczną sieć telefonii stacjonarnej (PSTN) albo z użyciem telefonu komórkowego, a innym razem poprzez telefon IP. Możliwość zdalnego uwierzytelnienia użytkownika jest dużą zaletą biometrii głosowej, bo już np. zdalne uwierzytelnienie przy pomocy odciska palca jest mocno utrudnione.

Wcześniejsze badania, z użyciem klasycznego podejścia opartego na modelach GMM, pokazywały, że zawężanie pasma sygnału lub kompresja stratna sygnału mowy obniża efektywność rozpoznawania mówcy. W moich badaniach jako metodę rozpoznawania postanowiłem wykorzystać inną technikę – hybrydową metodę SVM-GMM[6], która polega na klasyfikacji przy pomocy maszyny wektorów nośnych (SVM) superwerktorów złożonych z wartości średnich μ poszczególnych rozkładów normalnych mieszanego modelu Gaussa (GMM).

Cele badań, opisanych w [C6], były następujące:

[6] Campbell, W. M., Sturim, D. E., Reynolds, D. A.: Support vector machines using GMM supervectors for speaker verification. IEEE Signal Processing Letters, vol. 13, pp. 308–311 (2006)

- Porównanie zastosowanej metody SVM-GMM z wcześniejszą, klasyczną metodą weryfikacji mówcy opartą wyłącznie na modelach GMM z wykorzystaniem tzw. uogólnionego modelu mówcy, czyli modelu świata (ang. *universal background model*, UBM) dla różnych rodzajów transmisji.
- Porównanie wyników eksperymentów dla zadania weryfikacji z wynikami wcześniejszych badań dla zadania klasyfikacji[7].
- Wskazanie, które kodeki sygnału mowy byłyby najlepsze w celu utworzenia modeli mówców, które zapewniałyby najlepszą efektywność pracy systemów ASV niezależnie od warunków transmisji.

By przeprowadzić emulacje różnych rodzajów transmisji, użyłem różnych kodeków mowy, używanych w telefonii stacjonarnej (G.711), mobilnej (GSM 06.10, GSM 06.60), a także w telefonii IP (oprócz wymienionych kodeków, także G.723.1, G.729 i Speex). Dla porównania wykorzystałem też mowę niekodowaną. Analizowałem dwa przypadki: warunki dopasowane (ang. *matched conditions*), gdy modele głosów użytkowników zostały stworzone w tych samych warunkach, w których odbywały się testy, oraz warunki niedopasowane (ang. *mismatched conditions*), gdy sygnał mowy użyty do wyuczenia modeli głosów i sygnał mowy użyty do testów były transmitowane na różne sposoby (tj. z wykorzystaniem różnych kodeków).

Do testów użyłem nagrań pochodzących z bazy TIMIT – nagrania 200 mówców posłużyły do wyuczenia modeli UBM (które również są wykorzystywane w metodzie SVM-GMM), natomiast 430 mówców zostało użytych do eksperymentów z weryfikacją. Do stworzenia modeli mówców użyłem ok. 16 s mowy dla każdego użytkownika. Model SVM-GMM dla każdego użytkownika był tworzony w ten sposób, że w przestrzeni umieszczano osiem superwektorów pochodzących od danego mówcy oraz 200 superwektorów pochodzących od mówców użytych do stworzenia modelu UBM, którzy w ten sposób modelowali potencjalnych intruzów. Weryfikacji dokonywałem za pomocą krótkich, ok. dwusekundowych nagrań. Do testów zastosowałem protokół, w którym na każde pięć prób użytkownika uprawnionego przypadało pięć prób weryfikacji poprzez użytkownika nieuprawnionego (intruza).

Opublikowane w [C6] wyniki badań pokazały, że dla warunków dopasowanych efektywność weryfikacji mówcy jest w dużym stopniu skorelowana z jakością mowy oferowaną przez poszczególne rodzaje transmisji. Dlatego najniższy błąd (EER poniżej 3%) uzyskano dla mowy niekodowanej, zaś najwyższy (EER 5,40%) dla kodeka G.723.1, oferującego najniższą jakość mowy. Gdy testy były prowadzone w warunkach niedopasowanych, wówczas w zdecydowanej większości przypadków błąd się zwiększał, ale w sposób zróżnicowany. Gdy modele mówców stworzone przy pomocy kodeka G.723.1 były testowane sygnałem transmitowanym z użyciem kodeka GSM 06.60 (używanego np. w sieci UMTS), wówczas wzrost błędu EER był minimalny (tylko 0,27 pkt. %). Eksperymenty wykazały jednak, że niektóre pary kodeków „nie lubią się" – np. gdy modele były testowane przez kodek GSM 06.10, wówczas błąd EER rósł nawet dwu- lub trzykrotnie względem warunków dopasowanych. Oprócz tych wniosków, dodatkowo, analizując średnie wartości EER dla modeli mówców stworzonych przy pomocy różnych kodeków, wskazałem G.723.1 oraz Speex jako kodeki, które zapewniały średnio najniższy błąd EER, niezależnie od tego, jakim rodzajem sygnału mowy były testowane.

[7] Janicki, A., Staroszczyk, T.: Speaker recognition from coded speech using support vector machines. In: I. Habernal, V. Matoušek (eds.) Text, Speech and Dialogue, No. 6836 in Lecture Notes In Computer Science, pp. 291–298. Springer Berlin Heidelberg (2011)

Badania efektywności działania systemów rozpoznawania mówcy dla dźwięków nieartykułowanych

W artykule **[C7]** zamieściłem wyniki eksperymentów, które dotyczyły wpływu dźwięków nieartykułowanych (ang. *non-speech sounds*) na algorytmy rozpoznawania mówcy. Inspiracją do tych prac była obserwacja, którą poczyniłem podczas badań nad rozpoznawaniem mówcy z wykorzystaniem bazy TIMIT. Nagrania w tej bazie są posegmentowane i anotowane fonetycznie, tj. dostępna jest informacja, gdzie dokładnie znajduje się który fonem. Dzięki temu mogłem badać wpływ różnych fonemów i innych elementów akustycznych na efektywność rozpoznawania mowy. Podczas takich eksperymentów zauważyłem, że dźwięki nieartykułowane, takie jak oddechy, śmiech czy mlaśnięcia, również pomagają w procesie rozpoznawania mówcy.

Celem badań opublikowanych w [C7] było zweryfikowanie, jak bardzo informacja zawarta w dźwiękach nieartykułowanych przyczynia się do poprawności rozpoznawania mówcy. W tym celu przeprowadziłem eksperymenty z klasyfikacją mówców opartą na metodzie GMM-UBM, używając nagrań 430 mówców, każdorazowo wybierając jednak z sygnału tylko określone elementy akustyczne, np. samogłoski (oznaczone jako V), spółgłoski (oznaczone jako C), dźwięki nieartykułowane (oznaczone jako N) lub ich kombinacje (np. VC, VN, VCN). Badałem także sygnały o różnej jakości (mowa niekodowana, jakość telefonii stacjonarnej, jakość telefonii mobilnej).

Wyniki tych eksperymentów uważam za interesujące i do pewnego stopnia zaskakujące. Okazało się, że dla nagrań niekodowanych, zawierających jedynie dźwięki nieartykułowane (N) poprawność klasyfikacji wyniosła ponad 30%, co uznałem za zaskakująco wysokie. We wszystkich wypadkach nagrania z pełną zawartością akustyczną (VCN) osiągały lepsze wyniki niż nagrania zawierające właściwą mowę (VC), z tym że różnica ta była największa dla nagrań o najniższej jakości i wyniosła aż prawie 4 pkt. %. Wyniki zaprezentowane w [C7] pokazały też, że poprawność klasyfikacji mówcy spadała czasem o ponad 2 pkt. %, jeżeli modele mówców były uczone na mowie „czystej" (VC), a testowane na mowie rzeczywistej (VCN), czyli zawierającej np. oddechy.

Wnioskiem płynącym z artykułu [C7] jest stwierdzenie, że dźwięki nieartykułowane, takie jak oddechy czy śmiech, zawierają informację akustyczną, której nie powinno się pomijać. W praktyce może się to przekładać na sugestię, by nadmiernie nie „oczyszczać" (np. używając zbyt „agresywnego" detektora aktywności głosowej VAD) nagrań używanych w rozpoznawaniu mowy np. do tworzenia modeli mówców, by nie tracić ważnych informacji biometrycznych z sygnału mowy.

Badania dotyczące zabezpieczania systemów rozpoznawania mówcy przed atakiem poprzez odtworzenie nagrania

Błędy fałszywej akceptacji, występujące w systemach weryfikacji mówcy (ASV), mogą być również wprowadzane intencjonalnie. Dzieje się tak podczas ataków na systemy ASV polegających na podszywaniu się pod tożsamość innego użytkownika (ang. *spoofing*). Ataki takie mogą zostać przeprowadzone np. przy pomocy algorytmów konwersji głosu, przy pomocy syntezy mowy, w której modele akustyczne zostały wyuczone na nagraniach mówcy docelowego (ang. *target speaker*), czy wreszcie przy pomocy odtworzenia nagrania (ang. *replay attack*). Publikacje [C8, C9] dotyczą właśnie zagrożeń płynących z tych ataków oraz metod zabezpieczania systemów ASV przed tymi atakami.

Artykuł **[C8]** dotyczy tematu atakowania systemów ASV przy pomocy odtworzenia nagrania. Przedstawiłem w nim wyniki prac, rozpoczętych podczas stażu naukowego w ośrodku

EURECOM we Francji w 2014 r. we współpracy z N. Evansem i F. Alegre, polegających na ewaluacji zagrożeń atakami poprzez odtworzenia nagrania, porównywaniu tych zagrożeń z innymi atakami oraz poszukiwaniu zabezpieczeń (ang. *spoofing countermeasures*) przed atakiem poprzez odtworzenie nagrania.

Eksperymenty prowadziłem, używając sześciu różnych algorytmów weryfikacji mówcy. Oprócz wspominanych wcześniej metod GMM-UBM oraz SVM-GMM, użyłem również tych dwóch metod z dodaniem tzw. analizy czynnikowej (ang. *factor analysis*, FA), metody SVM-GMM z wykorzystaniem tzw. *nuissance attribute projection* (NAP) oraz metody tzw. wektorów tożsamości (ang. *i-vectors*), która w połączeniu z probabilistyczną liniową analizą dyskryminacyjną (ang. *probabilistic linear discriminant analysis*, PLDA) według aktualnego stanu sztuki zapewnia najlepsze wyniki weryfikacji mówcy.

Do badań wykorzystałem nagrania pochodzące z baz używanych w ewaluacjach systemów rozpoznawania mówcy (ang. *Speaker Recognition Evaluation*, SRE), organizowanych cyklicznie przez amerykański instytut NIST[8]. Wykorzystałem protokół testów, opracowany przez NIST dla celów ewaluacji SRE. Jednak podobnie do innych badań dotyczących zagrożeń atakami, zamiast prób mówców nieuprawnionych (ang. *impostor trials*) użyłem prób podszywania się pod mówców uprawnionych (ang. *spoofing trials*), w liczbie równej liczbie prób mówców uprawnionych (ang. *licit trials*), tj. 1352 prób dostępu.

W eksperymentach zastosowałem emulacje dziewięciu różnych wariantów środowiska, w którym może być potencjalnie przeprowadzony atak. Emulowałem trzy różne urządzenia odtwarzające nagranie (głośnik smartfonu, głośnik tabletu, głośnik wysokiej jakości), a także trzy różne warunki akustyczne (biuro, korytarz, komora bezechowa). Emulacje przeprowadziłem, wykorzystując odpowiednie odpowiedzi impulsowe.

Opublikowane w [C8] wyniki pokazały, że ataki przy pomocy nagrania mogą być niebezpieczne i bardziej skuteczne niż ataki przy pomocy syntezy mowy lub konwersji głosu, a są przy tym o wiele łatwiej dostępne niż zaawansowane technologie, takie jak konwersja mowy czy synteza mowy, której brzmienie można by dostosować do brzmienia docelowego mówcy. Badania pokazały, że kluczowa dla skuteczności ataku jest akustyka pomieszczenia – skuteczność ataku była najmniejsza przy emulacji korytarza z wyraźnym pogłosem (EER równy 24,5% dla systemu i-vectors-PLDA), a największa dla komory bezechowej (EER bliskie 50%, czyli atak jest prawie w 100% skuteczny). Na szczęście scenariusz odtwarzania nagranego studyjnie głosu w komorze bezechowej jest w praktyce mało możliwy. Charakterystyka częstotliwościowa urządzenia odtwarzającego dźwięk miała drugorzędne znaczenie, choć oczywiście głośnik wysokiej jakości umożliwiał nieznacznie większą skuteczność ataku. Ciekawą i niepokojącą obserwacją był znaczny wzrost błędu EER da systemów GMM-UBM oraz SVM-GMM, spowodowany przypuszczalnie niepożądanym działaniem algorytmu normalizacji.

W artykule [C8] zaprezentowałem również opis dwóch różnych metod wykrywania ataku poprzez nagranie. Jedna z nich to detektor odległego nagrania (ang. *Far-Field Detector*, FFD), druga zaś to metoda lokalnych wzorców binarnych (ang. *Local Binary Patterns*, LBP), zaproponowana w tym artykule po raz pierwszy w celu detekcji ataku poprzez odtworzenie nagrania. Pierwsza metoda, zaproponowana w roku 2011[9], polega na ekstrahowaniu z sygnału 12 parametrów, takich jak środek ciężkości widma czy wąskopasmowe indeksy modulacji. Druga metoda, początkowo zastosowana do detekcji mowy syntetycznej i konwertowanej[10], polega na użyciu specjalnych operatorów LBP, które wyznaczają tzw. wzorce binarne, działając

[8]NIST – National Institute of Standards and Technology, www.nist.gov

[9]Villalba, J., Lleida, E.: Preventing replay attacks on speaker verification systems. In: Proc. IEEE International Carnahan Conference on Security Technology (ICCST 2011), Barcelona, Spain, pp. 1–8 (2011)

[10]Alegre, F., Vipperla, R., Amehraye, A., Evans, N.: A new speaker verification spoofing countermeasure based on local binary patterns. In: Proc. Interspeech 2013, Lyon, France, (2013)

na „obrazie", który tworzą współczynniki kepstralne wyznaczone z kolejnych ramek sygnału. Pierwsza z metod (FFD) wykorzystuje fakt, że widmo sygnału nagrywanego z pewnej odległości (co zwykle ma miejsce podczas nagrywania kogoś bez jego wiedzy) ulega spłaszczeniu. Druga metoda (LBP), oparta jest na tym, że podczas analizy powtórnie zarejestrowanego sygnału mowy, co ma miejsce podczas ataku poprzez nagranie, pierwotna struktura wzorców binarnych ulega zaburzeniu. Według mojej najlepszej wiedzy, metoda LBP w tej pracy **została użyta po raz pierwszy** do detekcji ataku poprzez odtworzenie nagrania.

Prowadziłem eksperymenty z obiema omówionymi wyżej metodami, badając je zarówno w połączeniu z systemem ASV, jak i niezależnie od niego. Wyniki opublikowane w [C8] pokazały, że algorytm LBP zwraca lepsze wyniki detekcji (np. systemu i-vectors-PLDA i dla biura: EER = 9,5% dla LBP wobec EER = 13,6% dla metody FFD). Mimo że stanowi to znaczną poprawę względem systemu pozbawionego ochrony (EER dla zaatakowanego systemu wynosił 30%), to nadal detekcja tego rodzaju ataków obarczona jest znaczącym błędem i stanowi wyzwanie dla dalszych badań.

Badania dotyczące zabezpieczania systemów rozpoznawania mówcy przed atakami za pomocą syntezy i konwersji mowy

W artykule **[C9]** zaprezentowałem zaproponowaną przez siebie metodę wykrywania mowy syntetycznej i konwertowanej. Stanowi ona rozwinięcie **nowatorskiej metody detekcji nienaturalnej mowy**, którą zaproponowałem w roku 2015[11]. Polega ona na analizie błędu predykcji, który powstaje po odjęciu od sygnału oryginalnego sygnału, który daje się „przewidzieć" matematycznie metodami liniowej predykcji na podstawie poprzednich próbek. Inspiracją dla tej metody była obserwacja, że dla mowy nienaturalnej (czyli syntetycznej lub konwertowanej) sygnał mowy daje się przewidzieć:

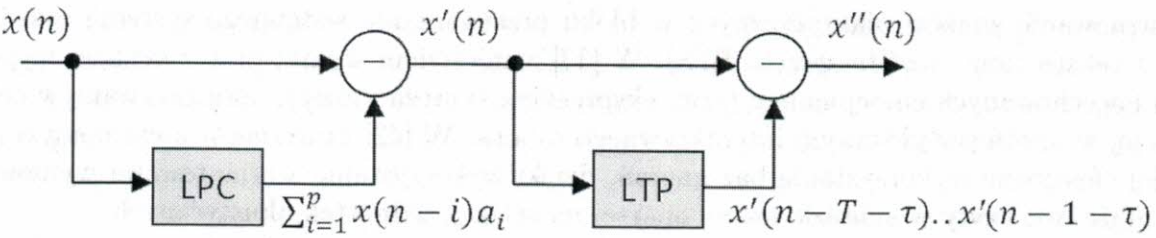
- czasami *zbyt łatwo*, ze względu na uproszczone modelowanie widma sygnału w algorytmach syntezy lub konwersji głosu, albo:
- czasami *zbyt trudno*, gdyż artefakty, które powstają na skutek niedoskonałości algorytmów syntezy i konwersji głosu, są trudne do przewidzenia technikami liniowej predykcji.

W zaproponowanej metodzie, nazwanej LPA (ang. *Linear Prediction Analysis*, analiza [błędu] liniowej predykcji), zastosowałem **odmienne podejście niż w klasycznych aplikacjach liniowej predykcji**. Zwykle dąży się do minimalizacji błędu predykcji, zaś większość użytecznej informacji zostaje wyekstrahowana z sygnału w postaci współczynników predykcji. W proponowanym podejściu współczynniki predykcji są w ogóle ignorowane, pełna uwaga zaś jest zwrócona w stronę sygnału błędu predykcji, zarówno krótkookresowej, jak i długookresowej (ang. *Long-Term Prediction*), użytych kaskadowo, jak pokazano na rysunku 2.4. Na podstawie tych sygnałów wyliczane są takie parametry jak: średni i maksymalny zysk predykcji krótkookresowej, średni i maksymalny zysk predykcji długookresowej, średnia i maksymalna energia błędu predykcji krótkookresowej itd. – w sumie 23 parametry.

Eksperymenty przeprowadziłem na bazie nagrań, udostępnionej przez organizatorów konkursu ASVspoof 2015 Challenge[12]. Parametry, wyliczone na podstawie sygnału błędu predykcji, były podawane na wejście algorytmu uczącego się. Eksperymentowałem z różnymi

[11] Janicki, A.: Spoofing Countermeasure Based on Analysis of Linear Prediction Error. In: Proc. Interspeech 2015, Dresden, Germany (2015)

[12] Wu, Z., Kinnunen, T., Evans, N., Yamagishi, J., Hanilçi, C., Sahidullah, M., Sizov, A.: ASVspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge. In: Proc. Interspeech 2015. Dresden, Germany (2015)



Rysunek 2.4: Schemat analizy sygnału mowy w metodzie LPA. $x(n)$ – wejściowy sygnał mowy, $x'(n)$ – sygnał błędu predykcji krótkookresowej, $x''(n)$ – sygnał błędu predykcji długookresowej.

algorytmami uczącymi się, najlepsze wyniki uzyskałem jednak dla algorytmu SVM z radialną funkcją jądra. Algorytm ten wytrenowałem na zbiorze uczącym, składającym się z ponad 16 tys. nagrań mowy naturalnej i nienaturalnej. Parametry pracy detektora dobierałem, wykorzystując podzbiór bazy ASVspoof zwany *Development*, zawierający cztery różne warianty syntezy lub konwersji mowy (pominąłem algorytm S2 ze względu na jego niską skuteczność). Testy prowadziłem zarówno na bazie *Development*, jak i bazie *Evaluation*, która zawierała prawie 200 tys. nagrań, zawierających mowę naturalną oraz nagrania pochodzące od 10 różnych wariantów syntezy lub konwersji mowy.

Wyniki zaprezentowane w artykule [C9] pokazały, że zaproponowana metoda umożliwia efektywną detekcję różnych wariantów mowy syntetycznej i konwertowanej. Dla bazy *Development* (bez algorytmu S2) uzyskałem poprawną detekcję ataków z błędem EER wynoszącym poniżej 1,3%. Średni EER dla 10 algorytmów z bazy *Evaluation* wyniósł 6,6%, gdyż zawierała ona algorytmy, które nie występowały w zbiorze uczącym.

W publikacji zamieściłem również porównanie zaproponowanej metody z opisaną wcześniej metodą LBP. Wyniki wykazały, że algorytm LPA zwraca niższe błędy detekcji niż algorytm LBP, zarówno dla algorytmów występujących w zbiorze uczącym, jak też dla algorytmów „nowych", nie widzianych podczas uczenia. Należy jednak zauważyć, że nagrania z bazy ASVspoof były krótkie, zaledwie kilkusekundowe, a więc kilkanaście razy krótsze niż występujące w bazie NIST, używanej w eksperymentach z atakiem za pomocą odtworzenia nagrania. Warto też zwrócić uwagę, że algorytm LBP osiągnął nieznacznie lepsze wyniki detekcji algorytmów S6, S7 i S9. Celowym wydaje się więc połączenie obu (lub więcej) metod w celu zwiększenia efektywności detekcji ataków i dalszego podnoszenia bezpieczeństwa systemów weryfikacji mówcy.

5 Omówienie pozostałych osiągnięć naukowo-badawczych

W niniejszym rozdziale zamieściłem informacje o osiągnięciach naukowo-badawczych dokonanych po uzyskaniu tytułu doktora, innych niż te, które stanowią osiągnięcie naukowe opisane w rodziałach 2–4. W przeważającej większości moje prace dotyczyły różnych aspektów przetwarzania sygnału mowy.

Po uzyskaniu tytułu doktora kontynuowałem przez pewien czas prace dotyczące syntezy mowy. W tym czasie opublikowałem prace, podsumowujące swoje osiągnięcia w dziedzinie podnoszenia jakości syntezy konkatenacyjnej dla języka polskiego [25, 46][13]. W artykule [44], zaprezentowanym na konferencji Interspeech 2005, przedstawiłem algorytm, który umożliwiał

[13]W tym rozdziale i w następnych odsyłacze dotyczą pozycji literatury zamieszczonych w wykazie dorobku w Załączniku 4.

rekonstruowanie znaków diakrytycznych w bloku przetwarzania wstępnego systemu systezy mowy z tekstu (ang. *text-to-speech*, TTS). W [33] zamieściłem wyniki prac z syntezą wypowiedzi nacechowanych emocjonalnie (tzw. ekspresyjna synteza mowy), opracowywaną w celu użycia jej w interfejsie głosowym interaktywnego robota. W [42] zaprezentowałem pomysł na bardziej efektywne wykorzystanie baz nagrań, dzięki wykorzystaniu wariantowości wymowy, który może być użyty w syntezie mowy opartej na selekcji jednostek akustycznych.

Z tematem klasycznej syntezy mowy łączy się zagadnienie wizyjnej syntezy mowy (ang. *visual speech synthesis*, VSS), która polega na generowaniu animacji z wizerunkiem twarzy, zsynchronizowanej z sygnałem mowy. Wraz z dyplomantami stworzyliśmy system „gadającej głowy" dla języka polskiego, nazwanej Karol, która wykorzystywała tzw. ramki kluczowe. System został opisany w [18, 41].

W ramach prac związanym z rozpoznawaniem mowy (ang. *Automatic Speech Recognition*, ASR), czyli zamianą sygnału mowy na tekst, wraz z dyplomantem opracowaliśmy grę komputerową „Rally Navigator" sterowaną głosem. System oparty na pakiecie CMU Sphinx umożliwiał rozpoznawanie komend w języku polskim z określonego zbioru, w sposób wystarczający do sterowania ruchem samochodu wyścigowego. System ten oraz wyniki przeprowadzonych eksperymentów zostały opisane w [16, 40]. W pracach dotyczących rozwoju systemów rozpoznawania mowy dla języka polskiego może być także przydatny zbiór nagrań, którego stworzenie zaprezentowałem w [19].

Za ciekawe badania uważam prace prowadzone w dziedzinie rozpoznawania dźwięków nieartykułowanych. W roku 2013 wziąłem udział w konkursie Social Signals Sub-Challenge w ramach Computational Paralinguistics Challenge (ComParE 2013). Zadaniem uczestników było rozpoznanie z jak największą dokładnością fragmentów sygnału mowy zawierających śmiech lub wypełnione przerwy (ang. *filled pauses*, czyli dźwięki typu „eeee", „yyyy"). Swoje wyniki zaprezentowałem na konferencji Interspeech 2013 [39]. Dzięki zaproponowanej hybrydowej metodzie opartej na modelowaniu GMM i SVM uzyskałem wynik, który okazał się być 3. wynikiem wśród przyjętych artykułów. Warto wspomnieć, że na całej konferencji Interspeech 2013, gromadzącej ponad tysiąc uczestników, byłem jedynym naukowcem z Polski prezentującym swoje badania.

W ramach badań nad przetwarzaniem mowy prowadziłem również prace nad rozpoznawaniem stanu emocjonalnego mówcy na podstawie analizy sygnału mowy. Efektem tych badań są publikacje [22, 45]. Artykuł [10] dotyczy porównania algorytmów rozpoznawania emocji zależnych i niezależnych od mówcy.

Kilka artykułów, które opublikowałem, dotyczyło zagadnienia badania jakości sygnału mowy. Ogólnie zagadnienie to jest przybliżone w artykule [20]. W [24] wraz z dyplomantami opisałem metodę badania wyrazistości transmisji w sieciach pakietowych z wykorzystaniem zdań nieprzewidywalnych semantycznie. W [21] przedstawiliśmy prace nad jednostronną metodą badania jakości, opartą na analizie obwiedni sygnału. Z kolei w [32] zaproponowaliśmy nowatorską metodę niwelowania wpływu strat pakietów na jakość transmitowanego sygnału mowy w telefonii internetowej. Za ważne publikacje uważam również prace [14, 35], dotyczące percepcji sygnału mowy przez osoby starsze.

Oprócz prac opisanych wcześniej, także kilka innych artykułów dotyczyło rozpoznawania mówcy. W [43] wraz z dyplomantem zaprezentowaliśmy trenowalny algorytm detekcji aktywności głosowej i wykazaliśmy jego wpływ na poprawność działania algorytmu rozpoznawania mówcy opartego na modelowaniu GMM. W [23] zaprezentowaliśmy system weryfikacji mówcy zależny od tekstu, w którym PIN wprowadzany z klawiatury był zastąpiony „PIN-em głosowym". W pracach [17, 31] wraz z dyplomantem opublikowałem wyniki eksperymentów z rozpoznawaniem mówcy z użyciem metody SVM-GMM, dla mowy o różnej jakości. Publika-

cje [38, 15] to inne prace z dziedziny bezpieczeństwa systemów weryfikacji mówcy, omówionej szerzej w rozdziale 4.2. Tematu bezpieczeństwa w szerszym zakresie, np. dotyczącego ukrywania informacji w transmisjach pakietowych czy badania zagrożeń w sieci Wi-Fi, dotyczyły kolejne publikacje [1, 5, 12, 13].

Do odmiennej grupy należą publikacje [11, 27]. Wraz ze współautorką (dr psychologii) zaprezentowaliśmy w nich przykład użycia algorytmów uczących się (w tym wypadku SVM) w rozpoznawaniu cech charakteru człowieka na podstawie analizy parametrów jego pisma odręcznego.