



**AGH**

**AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE**

**WYDZIAŁ INFORMATYKI, ELEKTRONIKI I TELEKOMUNIKACJI**

KATEDRA INFORMATYKI

Praca dyplomowa magisterska

*Strumieniowanie adaptacyjne danych multimedialnych z  
wykorzystaniem standardu DASH-MPEG*

Autor:

*Piotr Borowiec*

Kierunek studiów:

*Informatyka*

Opiekun pracy:

*dr inż. Łukasz Czekierda*

Kraków, 2014

*Oświadczam, świadomy(-a) odpowiedzialności karnej za poświadczenie nieprawdy, że niniejszą pracę dyplomową wykonałem(-am) osobiście i samodzielnie i nie korzystałem(-am) ze źródeł innych niż wymienione w pracy.*

*Serdecznie dziękuję ...*

## Spis treści

<b>1. Wprowadzenie</b>	7
<b>2. Zarys problemu</b>	9
<b>3. Przegląd istniejących rozwiązań</b>	11
3.1. Wstęp	11
3.2. Datagram Congestion Control Protocol	11
3.3. Real-time Transport Protocol i RTP Control Protocol	11
3.4. TCP Friendly	12
3.5. Podsumowanie	12
<b>4. Opis projektu i implementacji</b>	13
<b>5. Wdrożenie i testy</b>	15
<b>6. Podsumowanie</b>	17
<b>7. Bibliografia</b>	18



# **1. Wprowadzenie**

- Cel pracy - skąd się wziął pomysł, dlaczego praca jest tworzona
- Struktura - krótki opis dotyczący kolejnych rozdziałów
- Zakres merytoryczny - ramy dla projektu
- Wkład własny - krótki opis co się zrobiło i jak przebiegały prace.



## **2. Zarys problemu**

- Pokazanie jak zmienne warunki w sieci wpływają na strumieniowanie
- Wyjaśnienie na czym polegają problemy związane z implementacją gładkiego strumieniowania adaptacyjnego
- Przedstawienie istniejących rozwiązań związanych ze strumieniowaniem adaptacyjnym (wtyczka do VLC, dodatek do Youtube, wtyczka do Internet Explorera)





## **3. Przegląd istniejących rozwiązań**

### **3.1. Wstęp**

Istnieje wiele podejść do zagadnienia jakim jest strumieniowanie danych. Najbardziej popularnym spośród opisywanych poniżej jest para protokołów: Real-time Transport Protocol i RTP Control Protocol [2]. W niniejszym rozdziale opisane zostaną również protokoły: Datagram Congestion Control Protocol [4], TCP Friendly [6] oraz standard DASH-MPEG [7]. W każdym z powyższych przypadków zostaną zaprezentowane metody kontroli przeciążenia sieci i punktów końcowych transmisji.

### **3.2. Datagram Congestion Control Protocol**

Opis DCCP.

### **3.3. Real-time Transport Protocol i RTP Control Protocol**

RTP (Real-time Transport Protocol) pozwala na transport danych w systemach i aplikacjach czasu rzeczywistego wspierających interaktywne transmisje audio i video. Zwykle wykorzystuje UDP jako protokół transportowy, ale może działać nad innymi protokołami warstwy transportowej modelu ISO/OSI (DCCP, TCP - [2, 5]). Jeżeli sieć w której działa RTP wspiera multicasting to RTP pozwala na wysłanie danych do wielu odbiorców jednocześnie. RTP nie dostarcza mechanizmów QoS (Quality of Service), ani nie gwarantuje dostarczenia wysłanych danych na czas do odbiorcy.

Protokół RTCP (Real-time Control Protocol) stanowi protokół kontrolny dla RTP. Bazuje na okresowej transmisji pakietów kontrolnych do wszystkich uczestników sesji. Przekazuje informacje od odbiorców do nadawców na temat jakości transmisji. Pakiety RTCP zawierają identyfikator źródła (Canonical Name) oraz pozwalają na ustalenie liczby uczestników sesji, co pozwala na obliczenie z jaką częstotliwością należy wysyłać pakiety kontrolne.

RTP przenosi dane, które zwykle wymagają ustalonej przepustowości. Dzięki temu prawdopodobieństwo, że strumień RTP będzie systematycznie zajmował coraz większe pasmo jest niewielkie. Z drugiej strony, strumień nie może też zostać ograniczony bez uszczerbku na jakości transmisji danych, szczególnie jeżeli transmisja ma charakter interaktywny. Pasma potrzebne do sprawnej transmisji zależy od rodzaju przesyłanych danych. Rodzaj przesyłanych danych można identyfikować na podstawie pola

Payload Type w nagłówku pakietu RTP. Z każdym typem związany jest profil RTP opisujący parametry transmisji (w tym wymaganą przepustowość) [3].

### **3.4. TCP Friendly**

Opis TCP Friendly.

### **3.5. Podsumowanie**

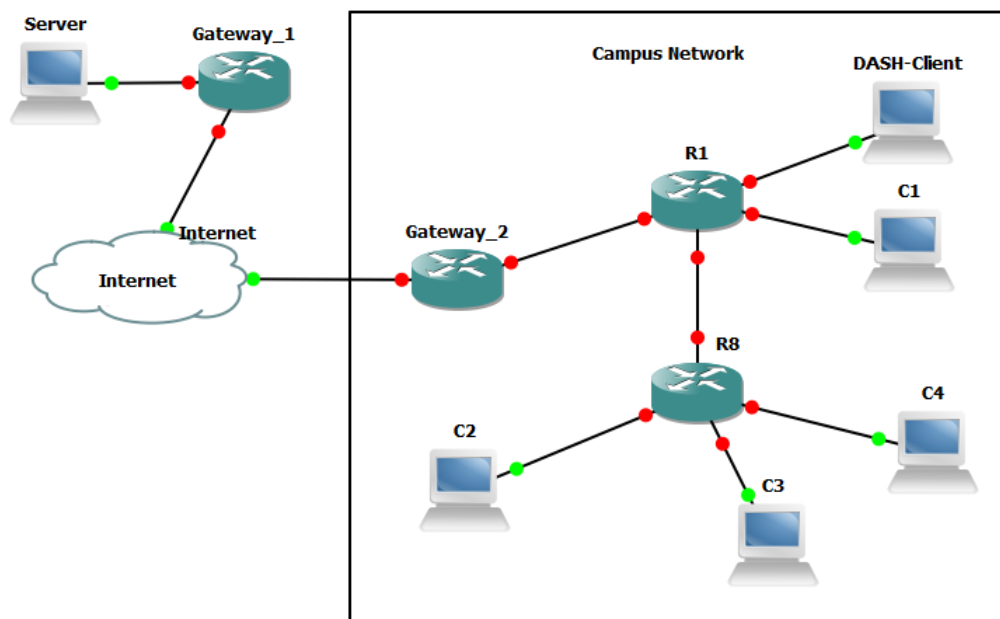
## **4. Opis projektu i implementacji**

- Krótka prezentacja algorytmu zaimplementowanego z artykułu
- Prezentacja własnego algorytmu lub poprawek do algorytmu z punktu poprzedniego
- Opis implementacji programu klienta (diagram klas, sekwencji...)
- Przedstawienie ewentualnych problemów przy implementacji i zastosowanych rozwiązań



## 5. Wdrożenie i testy

- Opis instalacji i uruchamiania programu klienta
- Opis przeprowadzonych testów działania/testów porównawczych
- Pokazanie w jaki sposób skonfigurować sieć lokalną (multicasty) w celu zmniejszenia jej obciążenia przy transmisjach jeden-do-wielu. Tylko klient DASH łączy się do serwera. Pozostałe komputery w sieci kampusowej odbierają transmisję od klienta DASH.





## **6. Podsumowanie**

- Podsumowanie pracy - co się udało, co się nie udało
- Podkreślenie wkładu własnego
- Dalsze możliwości rozwijania pracy





## Bibliografia

- [1] Guibin Tian, Yong Liu *Towards Agile and Smooth Video Adaptation in Dynamic HTTP Streaming*. Polytechnic Institute of New York University
- [2] Request For Comment 3550: RTP: A Transport Protocol for Real-Time Applications
- [3] Request For Comment 3551: RTP: RTP Profile for Audio and Video Conferences with Minimal Control
- [4] Request For Comment 4340: Datagram Congestion Control Protocol
- [5] Request For Comment 5762: RTP and the Datagram Congestion Control Protocol (DCCP)
- [6] Request For Comment 5348: TCP Friendly Rate Control (TFRC): Protocol Specification
- [7] Information technology — Dynamic adaptive streaming over HTTP (DASH) Part 1: Media presentation description and segment formats