

# BAB 1

## USULAN GAGASAN

### 1.1 Latar Belakang Masalah

Pada saat ini penggunaan internet menggunakan metode *IP Address*, yang merupakan barisan angka-angka yang berbeda disetiap perangkat seperti komputer dan ponsel untuk menamai perangkat tersebut. Dengan digunakannya *IP Address* dalam proses pengiriman data, masalah-masalah seperti skalabilitas, keamanan, dan efisiensi dalam pengambilan konten dapat muncul. Untuk mengatasi masalah tersebut, *Named Data Networking* (NDN) muncul sebagai paradigma baru yang bertujuan untuk mengembangkan arsitektur internet dalam jaringan yang sebelumnya *host-centric* menjadi *data-centric*. NDN memberikan pengenalan identitas kepada setiap konten yang dikirim melalui jaringan, berbeda dengan model tradisional IP yang memberikan identitas kepada perangkat/*host*. Dalam NDN, setiap paket yang dikirim mengandung sebuah pengenalan unik yang disebut "nama data" yang menjelaskan isi dari konten yang diinginkan, bukan alamat sumber atau tujuan paket tersebut. Dengan demikian, NDN memungkinkan pemrosesan dan pengambilan konten berdasarkan nama data yang diinginkan secara efisien, tanpa perlu mengetahui secara spesifik di mana data tersebut tersimpan dalam jaringan.

Namun pada NDN sendiri memiliki kelemahan dalam pengoperasiannya yang ada pada strategi *Forwarding* seperti kemacetan (*congestion*) pada *interface* dalam pengiriman data dan keterbatasan dalam penyimpanan data yang di-*cache* pada strategi *Caching*. Dengan adanya kebutuhan sumber daya jaringan yang melebihi kapasitas sumber daya jaringan menyebabkan terjadinya kemacetan (*congestion*) dalam strategi *Forwarding*. Pada strategi *Caching* sendiri juga terdapat masalah yaitu *content store* yang terbatas sehingga setiap kali data yang diminta di dalam *content store* tidak tersedia. Walaupun perkembangan dari NDN sudah banyak mengurangi masalah yang ada, nyatanya NDN masih memerlukan pengembangan yang lebih dalam, untuk mengatasi permasalahan yang telah dipaparkan maka solusi yang digunakan yaitu dengan *Reinforcement Learning* (RL).

## 1.2 Informasi Pendukung Masalah

Dalam *Named Data Networking*, Tanaka & Kawarasaki mengemukakan masalah yang ada kemudian mengusulkan 2 metode untuk menyelesaikannya. Solusi untuk strategi *Forwarding* yaitu mengontrol *interest packet* yang melewati *interface* dan pada strategi *Caching* yaitu menyesuaikan kebijakan *Caching*[10].

Menurut Qiang & Zhongli, *Reinforcement Learning* dapat digunakan untuk menyelesaikan masalah dengan mempelajari strategi tindakan yang optimal dalam berinteraksi dengan *environment*, sehingga sistem dapat memperoleh *reward* terbesar dari *environment*[25].

Watkins & Dayan mengatakan bahwa *Q-learning* merupakan algoritma dalam *Reinforcement Learning* yang bekerja dengan mempelajari keputusan optimal. Algoritma ini menghitung *Q-value* untuk setiap *action* dalam suatu keadaan dan memilih *action* dengan *Q-value* tertinggi sebagai tindakan yang optimal. Algoritma ini tidak memerlukan pengetahuan sebelumnya tentang *environment* dan dapat belajar melalui *trial and error* atau eksplorasi dan eksploitasi[7].

Menurut Mejri, S., Touati, H., & Kamoun, F., penelitian tentang kontrol kemacetan untuk jaringan NDN masih di tahap awal[19], lalu Narayan, A., dkk. mengembangkan sistem untuk menerapkan fungsi kontrol kemacetan yang kompleks dengan menempatkan kemacetan di luar jalur data[2]. Dengan mengintegrasikan *Q-learning*, Li, W., dkk. mengatakan bahwa *Reinforcement Learning* memungkinkan pengirim secara bertahap mempelajari kebijakan pengendalian kemacetan yang optimal[24].

Strategi *Forwarding* yang digunakan yaitu *Adaptive Smoothed RTT-based Forwarding* (ASF) merupakan strategi yang memilih *interface* terbaik berikutnya berdasarkan pengukuran RTT. Untuk memenuhi perubahan pada nilai RTT, Lehman, Vince., dkk menghitung nilai SRTT yang merupakan rata-rata dari nilai RTT dengan cara yang sama seperti TCP dan juga secara berkala memeriksa *interface* berikut lainnya untuk mempelajari RTT[22].

Iqbal & Asaduzzaman menggunakan beberapa algoritma *Reinforcement Learning* untuk memilih kebijakan *Caching* yang optimal dalam jaringan *Named*

*Data Networking* (NDN) seperti *Greedy*, *Epsilon Greedy*, dan *Upper Confidence Bound 1* (UCB 1)[18].

Pada *Least Recently Used* (LRU) ketika *cache* melebihi ukuran maksimumnya, ia akan menyingkirkan objek yang terakhir yang digunakan dan mengganti objek dengan yang baru menurut Arora, K., & Ch, D. R.[13] serta Martina, V., Garetto, M., & Leonardi, E.[23].

### **1.3 Analisis Umum**

#### 1.3.1 Aspek Keberlanjutan (*sustainability*)

Pada topik yang diusulkan permasalahan yang diambil yaitu masalah kemacetan (*congestion*) pada strategi *Forwarding* serta keterbatasan pada *content store* dalam penyimpanan data pada strategi *Caching*. Maka dari itu, *Reinforcement Learning* merupakan solusi yang dianggap tepat untuk keberlangsungan dalam arsitektur jaringan.

#### 1.3.2 Aspek Teknologi

Pada infrastruktur NDN saat ini terdapat beberapa strategi yang dapat diimplementasikan dalam topologi. Umumnya strategi yang digunakan adalah strategi *Forwarding* dan *Caching*, dimana strategi *Forwarding* kedepannya diharapkan dapat lebih meminimalisir hambatan dalam pengiriman data. Sedangkan pada strategi *Caching* diharapkan *interest packet* yang diminta oleh *user* dapat terkirim dengan cepat yang berasal dari *content store* dan data yang terdapat di dalam *content store* adalah data yang akan dibutuhkan oleh *user*. Pada topologi dengan dua strategi yang telah disebutkan dapat ditingkatkan efisiensinya dengan menggunakan *Reinforcement Learning*.

### **1.4 Kebutuhan yang Harus Dipenuhi**

Berdasarkan analisis yang telah dilakukan, aspek keberlanjutan (*sustainability*) dan aspek teknologi memiliki rencana sistem dengan menggunakan *Reinforcement Learning* pada dua mekanisme NDN, yaitu strategi *Forwarding* dan *Caching*. Rencana sistem yang diusulkan yaitu melakukan pelatihan secara berulang dengan melakukan eksplorasi dan eksploitasi (*trial and error*) dari elemen yang ada pada *Reinforcement Learning* dengan menggunakan data acuan yang diambil dari Mini-

NDN sesuai dengan topologi yang telah dirancang untuk memaksimalkan kinerja pada strategi yang ada pada NDN.

## 1.5 Solusi Sistem yang Diusulkan

*Reinforcement Learning* (RL) dapat memecahkan masalah yang sangat kompleks dengan melakukan *trial and error* dengan melakukan eksplorasi dan eksploitasi sehingga kemungkinan terjadinya kesalahan yang sama sangat kecil. Algoritma pada RL memiliki beberapa jenis, namun untuk solusi yang diusulkan menggunakan algoritma *Q-learning* dikarenakan merupakan salah satu algoritma dasar yang ada pada RL, dan salah satu metode pembelajaran berbasis nilai dengan tujuan utama memaksimalkan *reward* yang didapatkan dengan melakukan eksplorasi dan eksploitasi menggunakan kebijakan *epsilon-greedy*. Pada RL terdapat beberapa komponen seperti *agent* yang bertugas untuk membuat keputusan dalam melakukan *action*, *action* adalah suatu tindakan yang dilakukan oleh *agent*, *environment* yang merupakan tempat untuk *agent* dapat berinteraksi dalam melakukan *action*, *reward* adalah nilai untuk mengukur keberhasilan *action* dari *agent*, dan *policy* yang merupakan strategi kebijakan dalam menentukan *action* berikutnya. Bagian ini menjelaskan tentang usulan solusi yang ditawarkan yaitu dengan membuat *Intelligent NDN* untuk strategi *Forwarding* dan *Caching* menggunakan *Reinforcement Learning* yang dapat mengontrol kemacetan (*congestion*) serta keterbatasan dalam penyimpanan data yang di-cache. Pada bagian ini akan dijelaskan lebih lanjut mengenai karakteristik produk dan skenario penggunaan pada *Intelligent NDN*.

### 1.5.1 Karakteristik Produk

#### 1.5.1.1 Strategi *Forwarding* dengan *Reinforcement Learning*

- Fitur Utama

*Agent*: entitas yang membuat keputusan dalam pemilihan *interface* dengan nilai SRTT terendah serta memperbarui *Q-table* untuk melakukan *action* selanjutnya.

*Action*: suatu tindakan yang dilakukan oleh *agent* berupa strategi *Forwarding* dengan jenis *Adaptive Smoothed RTT-based Forwarding* (ASF) dalam memilih *interface* berdasarkan nilai SRTT terendah.

*Reward*: nilai untuk mengukur keberhasilan *action* dari strategi *Forwarding* dengan jenis *Adaptive Smoothed RTT-based Forwarding* (ASF) berupa terpilihnya *interface* dengan nilai SRTT yang terendah sehingga kemacetan (*congestion*) dapat teratasi.

*Policy*: strategi yang digunakan *agent* dalam menentukan *action* yang diambil oleh *agent* secara maksimal dengan memperhatikan tindakan sebelumnya menggunakan kebijakan *epsilon-greedy*.

- Fitur Dasar

*Environment*: lingkungan untuk *agent* dapat berinteraksi dalam pengiriman data berupa 5 buah *interface* dari pengoperasian topologi yang telah dirancang pada Mini-NDN.

- Sifat solusi yang diharapkan

Dari masalah yang telah dipaparkan dan fitur yang telah dirancang, solusi yang akan dicapai yaitu pengiriman data dapat terkirim dengan *interface* yang memiliki nilai SRTT terendah sehingga dapat mengontrol kemacetan (*congestion*).

#### 1.5.1.2 Strategi *Caching* dengan *Reinforcement Learning*

- Fitur Utama

*Agent*: entitas yang membuat keputusan dalam pemilihan *action* pada strategi *Caching* yaitu *cache* atau tidak di-*cache* atau *replacement Least Recently Used* (LRU) serta memperbarui *Q-table* untuk melakukan *action* selanjutnya.

*Action*: suatu tindakan yang dilakukan oleh *agent* berupa strategi *Caching* dalam menentukan data yang akan di-*cache* atau tidak di-*cache* atau *replacement Least Recently Used* (LRU) dengan nilai RTT yang sesuai.

*Reward*: nilai untuk mengukur keberhasilan *action* dari strategi *Caching* berupa ketepatan *agent* dalam memilih *action* (*cache*/tidak di-*cache*/*replacement LRU*) dengan nilai RTT yang sesuai.

*Policy*: strategi yang digunakan *agent* dalam menentukan *action* yang diambil oleh *agent* secara maksimal dengan memperhatikan tindakan sebelumnya menggunakan kebijakan *epsilon-greedy*.

- Fitur Dasar

*Environment*: lingkungan untuk *agent* dapat berinteraksi yaitu dengan menyediakan ukuran dari *content store* dengan batas maksimum 10.

- Sifat solusi yang diharapkan

Dari masalah yang telah dipaparkan dan fitur yang telah dirancang, solusi yang akan dicapai yaitu meningkatkan efisiensi penyimpanan data dalam *content store*.

#### 1.5.2 Skenario Penggunaan

Dalam skenario penggunaan *Intelligent NDN*, *stakeholder* yang memungkinkan untuk terlibat dalam penggunaan produk ini seperti perusahaan di bidang teknologi dan informasi khususnya dalam spesialis teknologi jaringan dan telekomunikasi. Selain itu juga *customer* dapat memanfaatkan *Intelligent NDN* dalam pengiriman data yang diinginkan.

##### 1.5.2.1 Strategi *Forwarding* dengan *Reinforcement Learning*

*Reinforcement Learning* dengan strategi *Forwarding* memiliki masalah yaitu kemacetan (*congestion*). Elemen yang ada pada *Reinforcement Learning* yaitu *environment* berupa *interface* untuk jalur pengiriman data dengan *agent* berupa entitas yang membuat keputusan dalam pemilihan *interface* dengan nilai SRTT terendah serta memperbarui *Q-table* untuk melakukan *action* selanjutnya yang dilakukan oleh strategi *Adaptive Smoothed RTT-based Forwarding* (ASF) dengan menggunakan kebijakan *epsilon-greedy*, kemudian *reward* didapatkan dengan terpilihnya *interface* yang memiliki nilai SRTT terendah.

##### 1.5.2.2 Strategi *Caching* dengan *Reinforcement Learning*

Permasalahan yang dihadapi oleh strategi *Caching* yaitu keterbatasan dalam penyimpanan data yang di-*cache* sehingga diperlukan *action* atau tindakan dalam *Reinforcement Learning* yang terdiri dari tiga *action* yaitu *cache*, tidak di-*cache*, dan *replacement Least Recently Used* (LRU) berdasarkan nilai RTT. Proses *action* dilakukan dengan menggunakan kebijakan *epsilon-greedy* pada *environment* berupa ukuran dari *content store*, dan *reward* didapatkan dari kesesuaian *action* yang dipilih berdasarkan nilai RTT.

## 1.6 Kesimpulan dan Ringkasan CD-1

Pada dokumen CD-1 membahas mengenai masalah serta solusi yang diusulkan dalam topik *Intelligent NDN* menggunakan *Reinforcement Learning*. Terdapat dua strategi NDN yang digunakan yaitu *Forwarding* dan *Caching*, dimana setiap strategi memiliki masalahnya masing-masing. Untuk strategi *Forwarding* memiliki masalah kemacetan (*congestion*), lalu untuk strategi *Caching* memiliki masalah yaitu keterbatasan dalam penyimpanan data yang di-*cache*. Penulis mengusulkan penggunaan *Reinforcement Learning* untuk menyelesaikan masalah yang ada dengan mencantumkan dua aspek yang terhubung yaitu aspek keberlanjutan (*sustainability*) dan aspek teknologi. Produk yang akan dibuat ditujukan untuk perusahaan di bidang teknologi dan informasi khususnya dalam spesialis teknologi jaringan dan telekomunikasi serta *customer* juga dapat memanfaatkan *Intelligent NDN* dalam pengiriman data yang diinginkan.