AI-Based Distributed Processing Automation in Digital Twin Network

draft-oh-nmrg-ai-adp-01

S-B. Oh (KSA), Y-G. Hong (Daejeon Univ.), J-S. Youn (DONG-EUI Univ),
HJ.Lee (ETRI), H-K Kahng (Korea Univ.)

nmrg Meeting@IETF 118 – Prague November 10. 2023
History and status

- 00 : draft-oh-nmrg-ai-adp-00 (July. 2023)
  - Title : Network management by automating distributed processing based on artificial intelligence
- 01 : draft-oh-nmrg-ai-adp-01 (Oct. 2023)
  - 1st presentation
  - Title : AI-Based Distributed Processing Automation in Digital Twin Network (To reflect the discussion result of IETF117 meeting)
Motivations

- Change of network complexity
  - High number of devices and data increase the network complexity
  - The possibility of malfunction or errors increases when network administrators/operators manage the network manually

- Adaptation to dynamically change network environment

- Maximizing the utilization of network resources
  - The increasing necessity of optimal resource allocation based on the characteristics of nodes providing network functions arises
Intention of this draft

• To efficiently adapt to the dynamically changing network environment
  • Find optimal configuration of systems using AI and DTN
  • Find optimal task distributed processing using AI and DTN
    • Requirements of task distributed processing
  • Propose automating distributed processing with DTN and AI
Conventional Task Distributed Processing Techniques and Problems

- Task Distributed Processing Technique
  • Distribute computational tasks among multiple nodes in a network

- Conventional techniques in task Distributed Processing
  • Load balancing
  • Parallel processing
  • Pipelining

- Challenges and problems in Task Distributed Processing
  • Prevention of Single network node overload
  • Prevention of overall process delays caused by bottleneck
  • Prevention of entire process disruption caused by network node failure
Requirements of Task Distributed Processing

- Scalability
  - The ability to add or remove nodes from the network and distribute tasks efficiently and effectively, without compromising performance or functionality.

- Fault tolerance
  - The ability to handle node failures and network outages without disrupting overall system performance or task completion.

- Load balancing
  - The ability to distribute tasks evenly across all nodes, ensuring that no single node becomes overwhelmed or underutilized.

- Task coordination
  - The ability to manage task dependencies and ensure that tasks are completed in the correct order and on time

- Resource management
  - The ability to manage system resources such as memory, storage, and processing power effectively, to optimize task completion and minimize delays or errors.

- Security
  - The ability to ensure the integrity and confidentiality of data and tasks and protect against unauthorized access or tampering.
Automating Distributed Processing with Digital Twin and AI

- DT and AI technology for Real-time Task Distribution:
  • The real-time updates from digital twin network enable continuous, optimal task distribution.
  • AI algorithms analyze network conditions and user demand in real-time.
  • Enables dynamic task distribution and processing based on current network conditions.

- Automatic Task Rerouting:
  • The system automatically reroutes tasks to less congested network areas.
  • Reduces delays and enhances overall performance.

- AI-driven Task Allocation:
  • AI algorithms, based on digital twin data, can automatically optimize network operations.
  • Tasks are distributed to nodes based on factors like processing power and available memory.

- Data that AI models can utilize within the digital twin network
  • Network data
  • Task and task characteristic data
  • Performance and resource data
  • Network configuration and device data
An example of AI system for Task Distributed Processing

- **Input layer**
  - Size of task ($m$)
  - Number of components ($n$)
  - Division resolution ($\gamma$)
  - Network status ($k$)
  - Available computing resources ($r$)
  - Distance ($\lambda$)

- **Hidden layer**
  - Two layers for each DNN model
  - Each hidden layer consists of 128 neurons

- **Output layer**
  - Optimal offloading ($o^*$)
  - Optimal partitioning ($z^*$)
Thanks!!

Questions & Comments