گزارش شماره ۲ بر اساس مقاله ارسالی

تهیه کننده : محمد کدخدائی

Deep reinforcement learning-based optimal deployment of IoT machine learning jobs in fog computing architecture

تاریخ	فعالیت
۸ خرداد ۱۴۰۴	مصاحبه فنی
۱۷ خرداد ۱۴۰۴	دریافت مقاله
	بررسی و مطالعه مقاله، ترجمه قسمتهای مهم
۱۳ تیر ۱۴۰۴	ارسال گزارش شماره ۱
۱۸ تیر ۱۴۰۴	ارائه آنلاین (گوگل میت)
	مطالعه مراجع اشاره شده در مقاله
	شماره مقالات بررسی شده به ترتیب دانلود و مطالعه : ۴، ۲۰، ۲۶، ۲۹، ۱۰، ۲۸، ۲۳، ۲۷
۶ مرداد ۱۴۰۴	شروع تبدیل مقاله به سورس کد پایتون
	تهیه لیست ثابت‌ها و پارامترهای مقاله
۸ مرداد ۱۴۰۴	تهیه گزارش شماره ۲
	بررسی روش پیاده سازی ابعاد Jobs و Infrastructure مطابق مقاله برای شکل دهی State

Constants based on the paper's parameters
File: Code/Constants_with-comment.py

```python
# By: Mohammad Kadkhodaei # 1404-05-06 # Constants based on the paper's parameters # Job and
simulation parameters (from Section 6.2.1, Table 4) NUM_JOBS = 100 # Number of jobs to simulate (from
profile manager) NUM_EPISODES = 120 # Number of training episodes (from profile manager) NUM_SENSORS =
21 # Total IoT sensors in environment (from Table 5) # Infrastructure node counts (from Section 6.2.1)
NUM_FOG_NODES = 7 # Intermediate fog nodes (Raspberry Pi 4B setups) NUM_EDGE_NODES = 14 # Edge nodes
(single-core devices) NUM_CLOUD_NODES = 1 # Single cloud node with elastic resources # Resource
demands (from Table 4) MIN_vCPU_PER_SENSOR = 0.01 # Minimum vCPU demand per sensor (random range)
MAX_vCPU_PER_SENSOR = 0.1 # Maximum vCPU demand per sensor (random range) MIN_MEM_PER_SENSOR = 0.01 #
Minimum memory demand per sensor in GB (random range) MAX_MEM_PER_SENSOR = 0.1 # Maximum memory demand
per sensor in GB (random range) vCPU_PER_FL_SERVER = 0.0001 # vCPU demand for FL server per task
MEM_PER_FL_SERVER = 0.001 # Memory demand for FL server per task in GB FL_EPOCHS = 20 # Number of FL
training epochs MODEL_SIZE = 5 # Size of FL model parameters in MB FL_TRAINING_FREQ = 0.6 # Fraction
of time FL process needs to restart # Infrastructure resource capacities (from Table 5) CLOUD_vCPU =
float('inf') # Cloud node has elastic vCPU capacity CLOUD_MEM = float('inf') # Cloud node has elastic
memory capacity FOG_vCPU = 4 # Fog nodes have 4 vCPUs (ARM Cortex-A72) FOG_MEM = 4 # Fog nodes have
4GB RAM EDGE_vCPU = 1 # Edge nodes have 1 vCPU EDGE_MEM = 1 # Edge nodes have 1GB RAM # Computation
costs (from Table 5, based on AWS Fargate pricing) CLOUD_vCPU_COST = 0.97152 # $/vCPU/Day for cloud
nodes CLOUD_MEM_COST = 0.10668 # $/GB/Day for cloud nodes FOG_vCPU_COST = 0.4 # $/vCPU/Day for fog
nodes FOG_MEM_COST = 0.05 # $/GB/Day for fog nodes EDGE_vCPU_COST = 0.0 # Edge nodes have no
computation cost EDGE_MEM_COST = 0.0 # Edge nodes have no memory cost # Data transfer costs (from
Section 6.2.1) TYPE1_COST = 0.09 # $/GB for cloud outbound traffic (to internet) TYPE2_COST = 0.32 #
$/GB for internet traffic (from providers) TYPE3_COST = 0.16 # $/GB for intranet traffic (half of type
2) # DDPG algorithm parameters (from Table 4 and Section 4) ACTOR_LR = 0.0005 # Learning rate for
actor network CRITIC_LR = 0.002 # Learning rate for critic network GAMMA = 0.99 # Discount factor for
future rewards TAU = 0.005 # Soft update parameter for target networks BUFFER_SIZE = 100000 # Replay
buffer size BATCH_SIZE = 256 # Mini-batch size for training NOISE_STD = 0.1 # Initial standard
deviation for exploration noise NOISE_DECAY = 0.9995 # Decay rate for exploration noise # Task
creation parameters SEGMENT_SIZE = 2 / (NUM_SENSORS * 10)
```