

```

%% Summary
% Service Image, 서비스별 우선순위, 서비스별로 처리에 필요한 리소스 수를
결정
% 서비스 Image(S) : MEC 는 S 라는 서비스를 제공하기 위해서 해당하는
이미지를 가지고 있어야 함
% (참고) Request(R) : 사용자는 R 이라는 서비스에 해당하는 요청을 보낼 수 있음

%% Init
clc;clear;close all;

addpath("../Z-common");

%% System Param

% 생성되는 파일을 저장할 하위 디렉토리
subdir = 'save/';

% 시나리오 ID 구성 : X_Y_Z_AAAA
% X : N_NODE_PER_ROW 를 의미
% Y : DELAY_MAX 를 의미(DELAY_MIN=0 으로 고정)
% Z : MEC 를 생성할 확률값
% AAAA ; 임의로 붙인 순서번호
% scenario_id = '4_1.0_0.3_0001'; % ILP/LR subopt 확인용 시나리오
%scenario_id = '5_1.0_0.3_0001'; % 작은 네트워크 실험용
scenario_id = '10_1.0_0.3_0001'; % Large 네트워크 실험용

% 읽어올 파일의 경로/이름 지정
prevDataDir = '../A-NetworkSetup/';
prevDataFullPath = strcat(prevDataDir, subdir, scenario_id, '.mat');
load(prevDataFullPath);

% 코드를 실행하면 여러개의 파일이 생성되는데, 파일의 이름 기본값 설정
out_filename = scenario_id; % 저장용 파일의 이름의 기본값을 설정

% 서비스 종류는 인덱스로 구분하고 1,2,3,4, ... 의 ID 를 가짐
%SERVICE_INDEX_MAX = 5; % X=4,5 인 경우
SERVICE_INDEX_MAX = 10; % X=10 인 경우

```

```
% MEC 에 각 SVC IMG 가 설치되어 있는지 여부를 결정할 확률값
%svc_image_install_prob_max = 0.5; % X=4,5 인 경우
svc_image_install_prob_max = 0.3; % X=10 인 경우
```

```
MIN_PerSVC_REQ_SZ = 1; % 서비스별, 하나의 req 의 크기
MAX_PerSVC_REQ_SZ = 3; % 서비스별, 하나의 req 의 크기
MIN_PRI_LIMIT = 1; % 서비스별, 우선순위 최솟값
MAX_PRI_LIMIT = 3; % 서비스별, 우선순위 최대값
```

```
MEC_CAPA_MIN = 5;
MEC_CAPA_MAX = 15;
```

```
% 터미널 출력값을 텍스트 형태로 txt 파일에 저장
% "arguments must contain a char vect..." 라는 오류 나오면
% 파일명을 구성하는 모든 텍스트를 ""가 아닌 "로 사용하기
diary_filename = strcat(subdir, out_filename, '.txt');
if( isfile(diary_filename) )
    delete(diary_filename);
end
diary(diary_filename); % 시작!
```

```
%% 각 MEC 에 설치된 서비스 이미지 결정하기
% 확률적으로 결정하면 되지뭐...
% 각 MEC 는 최대 SERVICE_INDEX_MAX 만큼의 이미지를 설치했을 수 있다.
% 그 보다 작은 경우가 대부분이겠지만...
fprintf("Setup: pre-installed svc images to each MEC\n");
for i=1:1:total_mec_count
    % 클래스의 setter 를 호출할땐, 인스턴스를 덮어쓰기!!!
    mecs(i) = mecs(i).setInitialize(SERVICE_INDEX_MAX, MEC_CAPA_MIN,
    MEC_CAPA_MAX);
    mecs(i) = mecs(i).setInstalledSvc(svc_image_install_prob_max);
    while( sum(mecs(i).installed_svc_msk) <= 0 )
        % 최소한 하나의 서비스는 처리할 수 있도록!
        mecs(i) = mecs(i).setInstalledSvc(svc_image_install_prob_max);
    end
end
clear i
```

```

% 결과 출력하기
mat = zeros(total_mec_count, SERVICE_INDEX_MAX);
for i=1:1:total_mec_count
    mat(i,:) = mecs(i).installed_svc_msk';
end
fprintf('Result: installed svc imgs for each mec\n');
mat
clear i
clear mat

%% 각 서비스별로 결정할 것 들...
service = ServiceInfo(SERVICE_INDEX_MAX, MIN_PerSVC_REQ_SZ,
MAX_PerSVC_REQ_SZ, MIN_PRI_LIMIT, MAX_PRI_LIMIT);

%% Finalizing
fprintf("Diary... off\n");
diary off % txt 파일에 터미널 출력값 저장(diary) 종료

fprintf("Saving to %s.mat\n", out_filename); % mat 파일에 데이터 저장
save(strcat(subdir, out_filename, ".mat")); % workspace 의 변수를 mat 파일로 저장

```