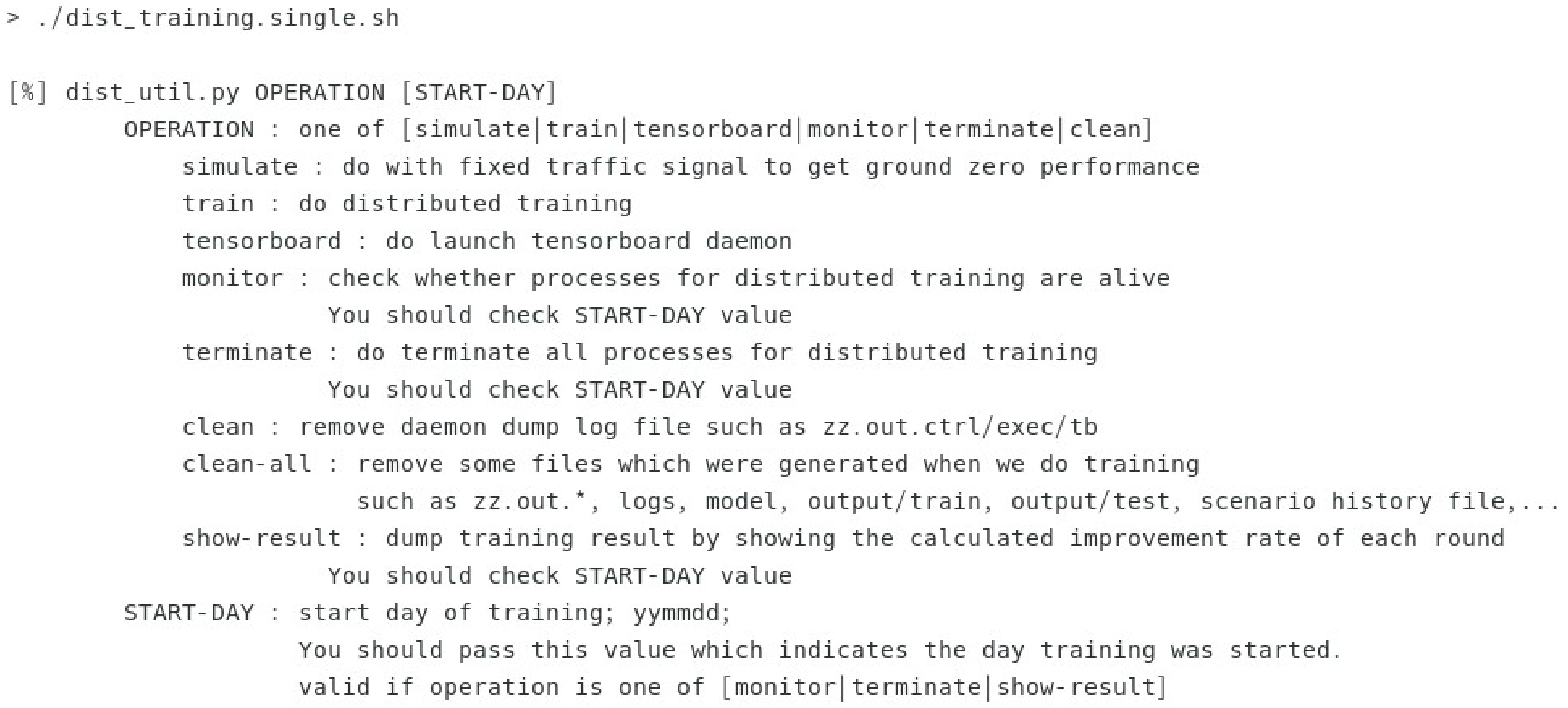
dist_training.single.sh를 이용한
축적 훈련 실험 방법

2022. 08. 12.

dist_training.single.sh

- 단일 노드에서 다수의 프로세서를 이용한 분산 훈련을 할수 있도록 하기 위한 스크립트
  - 고정 신호 실험, 훈련, 텐서보드 가동, 실험 프로세스 모니터링, 실험 종료, 실험 중간 파일 삭제, 실험 결과 보기 등을 제공
  - 다수의 노드를 이용한 분산 훈련도 가능



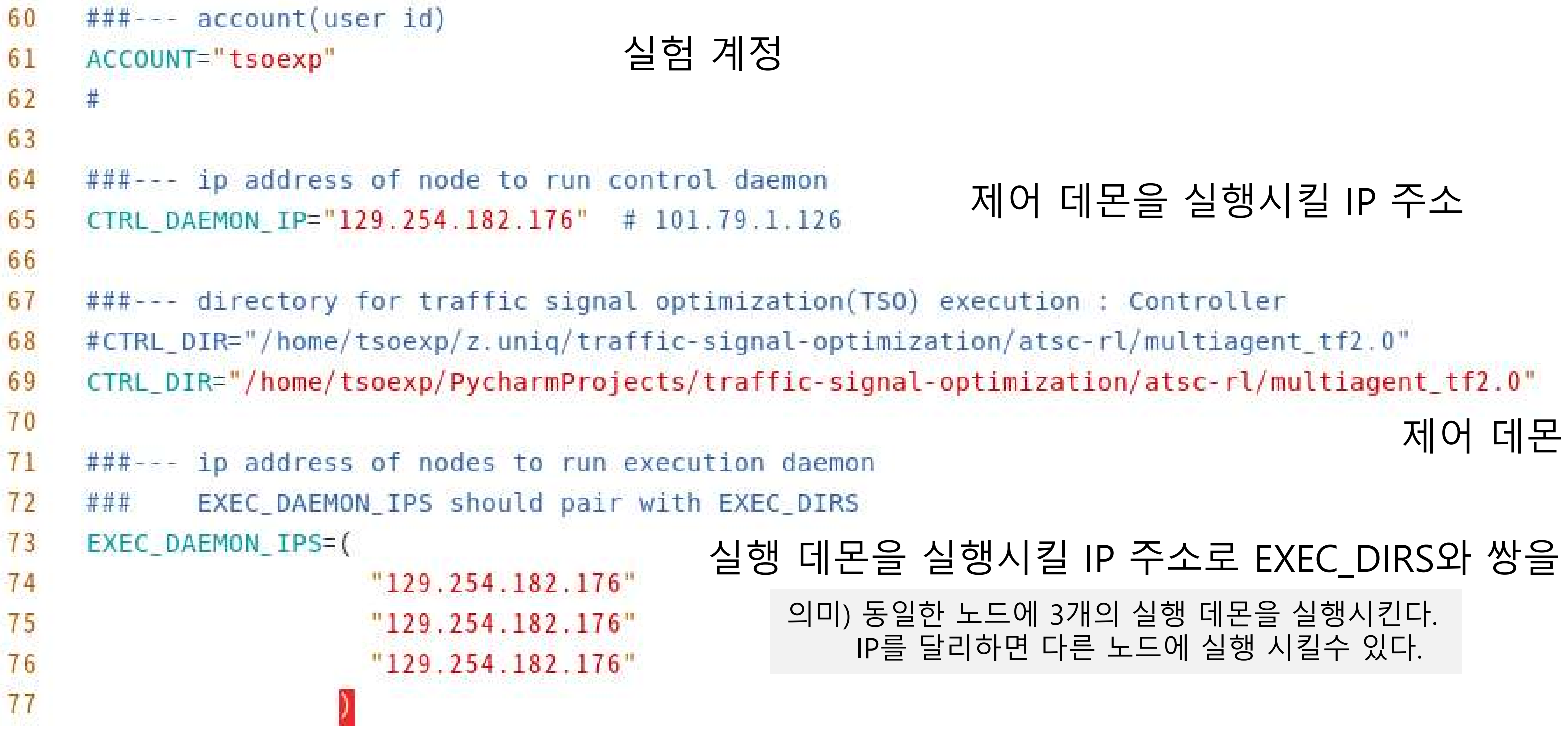
```
> ./dist_training.single.sh

[%] dist_util.py OPERATION [START-DAY]
        OPERATION : one of [simulate|train|tensorboard|monitor|terminate|clean]
            simulate : do with fixed traffic signal to get ground zero performance
            train : do distributed training
            tensorboard : do launch tensorboard daemon
            monitor : check whether processes for distributed training are alive
                      You should check START-DAY value
            terminate : do terminate all processes for distributed training
                      You should check START-DAY value
            clean : remove daemon dump log file such as zz.out.ctrl/exec/tb
            clean-all : remove some files which were generated when we do training
                        such as zz.out.*, logs, model, output/train, output/test, scenario history file,...
            show-result : dump training result by showing the calculated improvement rate of each round
                      You should check START-DAY value
        START-DAY : start day of training; yymmdd;
                    You should pass this value which indicates the day training was started.
                    valid if operation is one of [monitor|terminate|show-result]
```

환경 설정

- 시뮬레이터와 최적화 프로그램이 동작할 수 있도록 필요한 프로그램/라이브러리 설치
  - 참고 : https://etri.gov-dooray.com/project/posts/3312471706655197412

- 패스워드 없이 SSH 로그인이 가능하도록 설정
  - ssh key 생성 & 복사
  - 참고 : sshKeyGenAndCopy.sh

스크립트 이용한 실험 설정 : 실험 계정, IP 주소, 프로그램 위치



```
###--- account(user id)
ACCOUNT="tsoexp"
#

###--- ip address of node to run control daemon
CTRL_DAEMON_IP="129.254.182.176"  # 101.79.1.126

###--- directory for traffic signal optimization(TSO) execution : Controller
#CTRL_DIR="/home/tsoexp/z.uniq/traffic-signal-optimization/atsc-rl/multiagent_tf2.0"
CTRL_DIR="/home/tsoexp/PycharmProjects/traffic-signal-optimization/atsc-rl/multiagent_tf2.0"

###--- ip address of nodes to run execution daemon
###     EXEC_DAEMON_IPS should pair with EXEC_DIRS
EXEC_DAEMON_IPS=(
                  "129.254.182.176"
                  "129.254.182.176"
                  "129.254.182.176"
                )
```

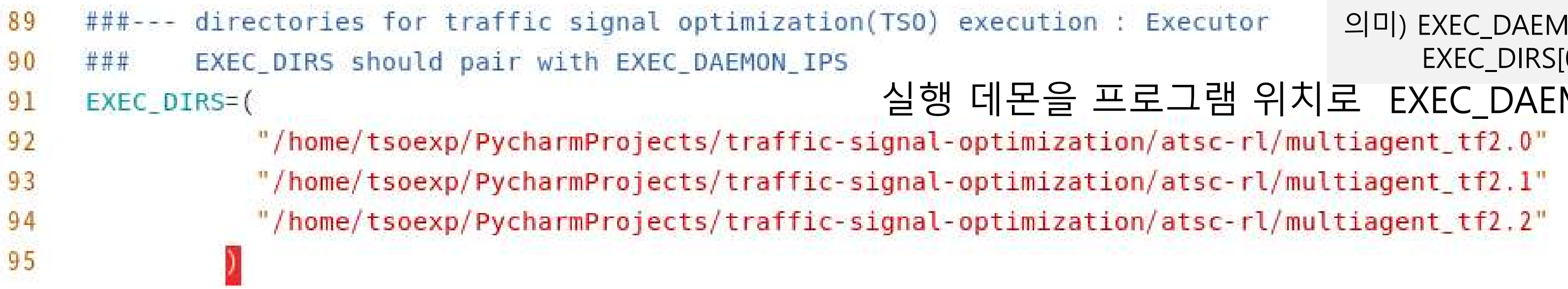
실험 계정

제어 데몬을 실행시킬 IP 주소

제어 데몬을 프로그램 위치

실행 데몬을 실행시킬 IP 주소로 EXEC_DIRS와 쌍을 이룸

의미) 동일한 노드에 3개의 실행 데몬을 실행시킨다.
IP를 달리하면 다른 노드에 실행 시킬수 있다.



```
###--- directories for traffic signal optimization(TSO) execution : Executor
###     EXEC_DIRS should pair with EXEC_DAEMON_IPS
EXEC_DIRS=(
           "/home/tsoexp/PycharmProjects/traffic-signal-optimization/atsc-rl/multiagent_tf2.0"
           "/home/tsoexp/PycharmProjects/traffic-signal-optimization/atsc-rl/multiagent_tf2.1"
           "/home/tsoexp/PycharmProjects/traffic-signal-optimization/atsc-rl/multiagent_tf2.2"
         )
```

의미) EXEC_DAEMON_IPS[0]의 실행 데몬 프로그램 위치는
EXEC_DIRS[0]

실행 데몬을 프로그램 위치로 EXEC_DAEMON_IPS와 쌍을 이룸

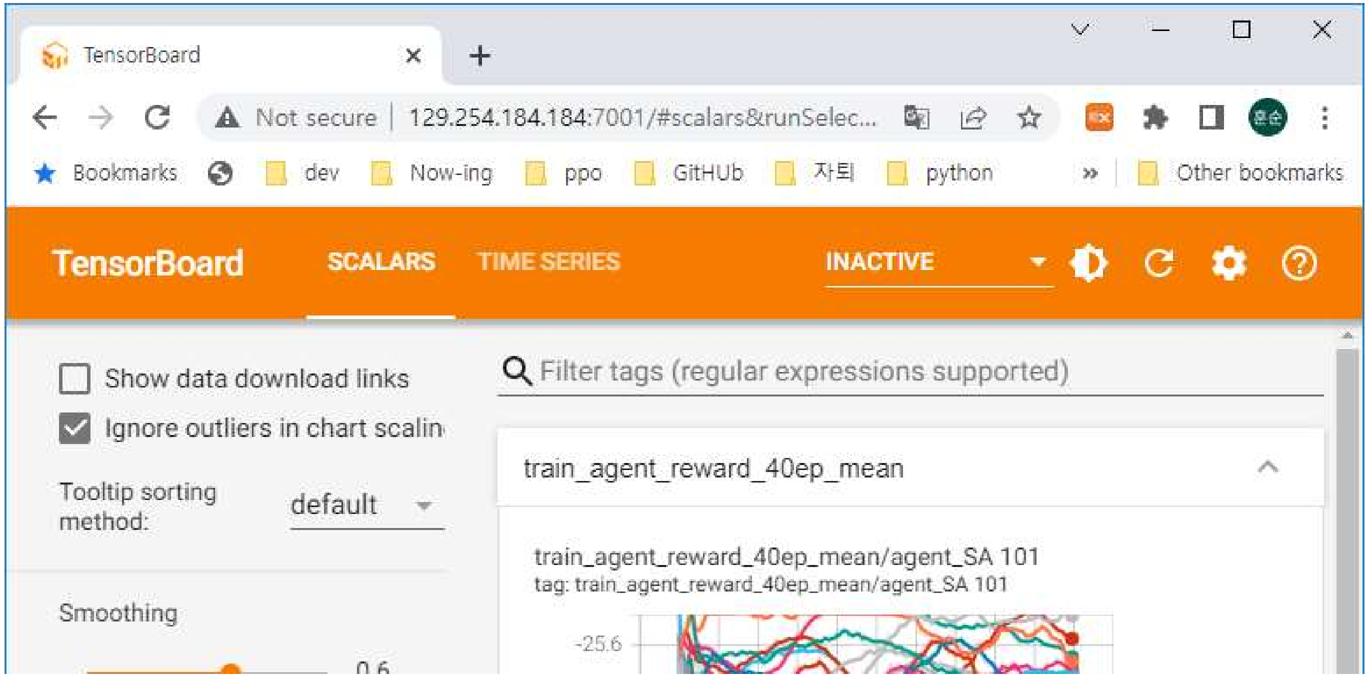
스크립트 이용한 실험 설정 : 통신 포트, 텐서보드 포트 설정

```
###--- port to communicate btn ctrl daemon and exec daemon
PORT=2727 #2727 3001  3101  3201  3301

###--- port for tensorboard
###--- TB_PORTS should pair with EXEC_DAEMON_IPS
TB_PORTS=(
                6006
                6016
                6026
        )
```

제어 데몬과 실행 데몬의 메시지 교환을 위한 통신 포트

텐서보드 접근을 위한 포트로 EXEC_DAEMON_IPS와 쌍을 이룸

의미) EXEC_DAEMON_IPS[0] 관련 TB를 위한 포트는 TB_PORTS[0]인 6006 이다.
따라서, http://EXEC_DAEMON_IPS[0]:TB_PORTS[0] 으로 접근 가능



☐ Show data download links

☑ Ignore outliers in chart scalin

Tooltip sorting method: default

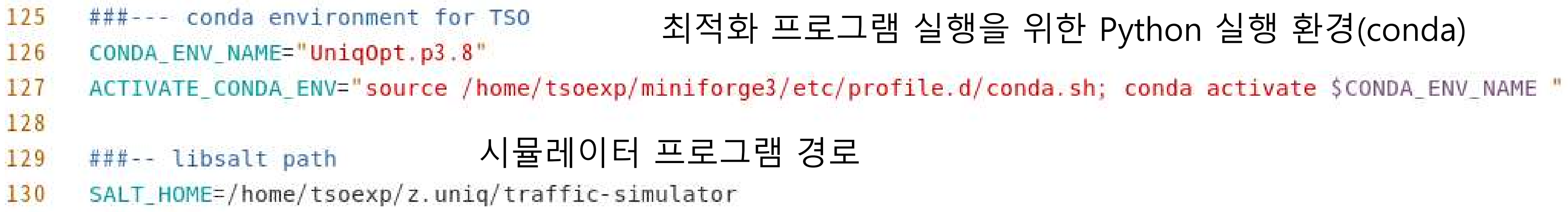
Smoothing

Filter tags (regular expressions supported)

train_agent_reward_40ep_mean

train_agent_reward_40ep_mean/agent_SA 101
tag: train_agent_reward_40ep_mean/agent_SA 101

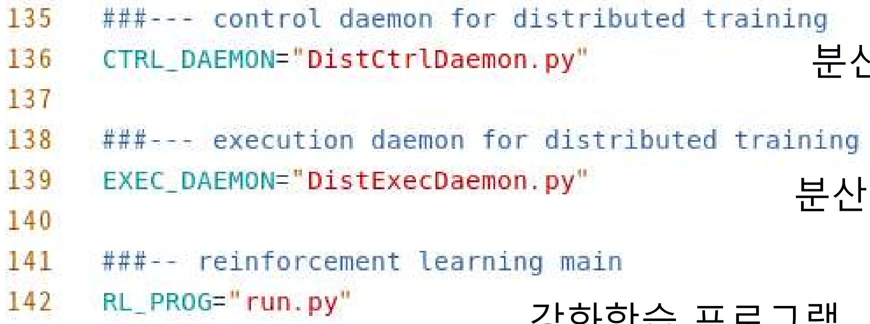
스크립트 이용한 실험 설정 : 최적화 실행 환경, 시뮬레이터 경로, 실행 프로그램



```
###--- conda environment for TSO
CONDA_ENV_NAME="UniqOpt.p3.8"
ACTIVATE_CONDA_ENV="source /home/tsoexp/miniforge3/etc/profile.d/conda.sh; conda activate $CONDA_ENV_NAME "

###-- libsalt path
SALT_HOME=/home/tsoexp/z.uniq/traffic-simulator
```

최적화 프로그램 실행을 위한 Python 실행 환경(conda)

시뮬레이터 프로그램 경로



```
###--- control daemon for distributed training
CTRL_DAEMON="DistCtrlDaemon.py"

###--- execution daemon for distributed training
EXEC_DAEMON="DistExecDaemon.py"

###-- reinforcement learning main
RL_PROG="run.py"
```

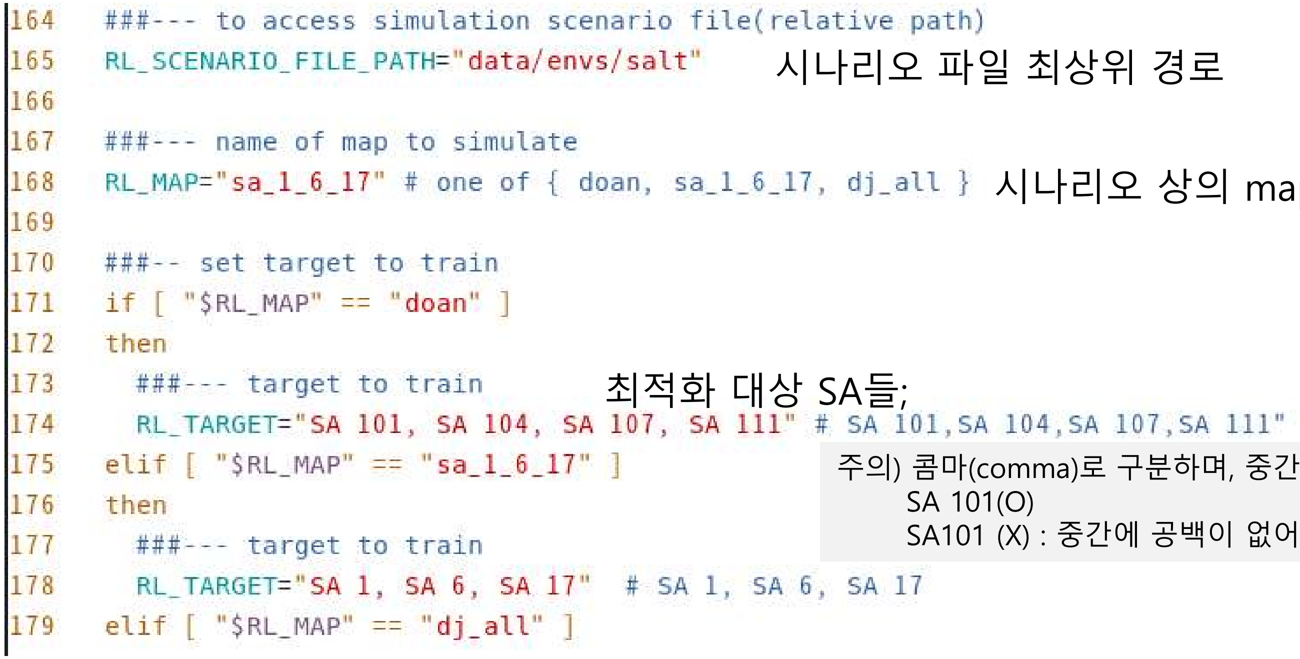
분산 훈련을 위한 제어 데몬

분산 훈련을 위한 실행 데몬

강화학습 프로그램

스크립트 이용한 실험 설정 : (강화학습 파라미터) 시나리오

제어 데몬 프로그램 실행 인자



```
###--- to access simulation scenario file(relative path)
RL_SCENARIO_FILE_PATH="data/envs/salt"

###--- name of map to simulate
RL_MAP="sa_1_6_17" # one of { doan, sa_1_6_17, dj_all }

###-- set target to train
if [ "$RL_MAP" == "doan" ]
then
  ###--- target to train
  RL_TARGET="SA 101, SA 104, SA 107, SA 111" # SA 101,SA 104,SA 107,SA 111"
elif [ "$RL_MAP" == "sa_1_6_17" ]
then
  ###--- target to train
  RL_TARGET="SA 1, SA 6, SA 17"  # SA 1, SA 6, SA 17
elif [ "$RL_MAP" == "dj_all" ]
```

시나리오 파일 최상위 경로

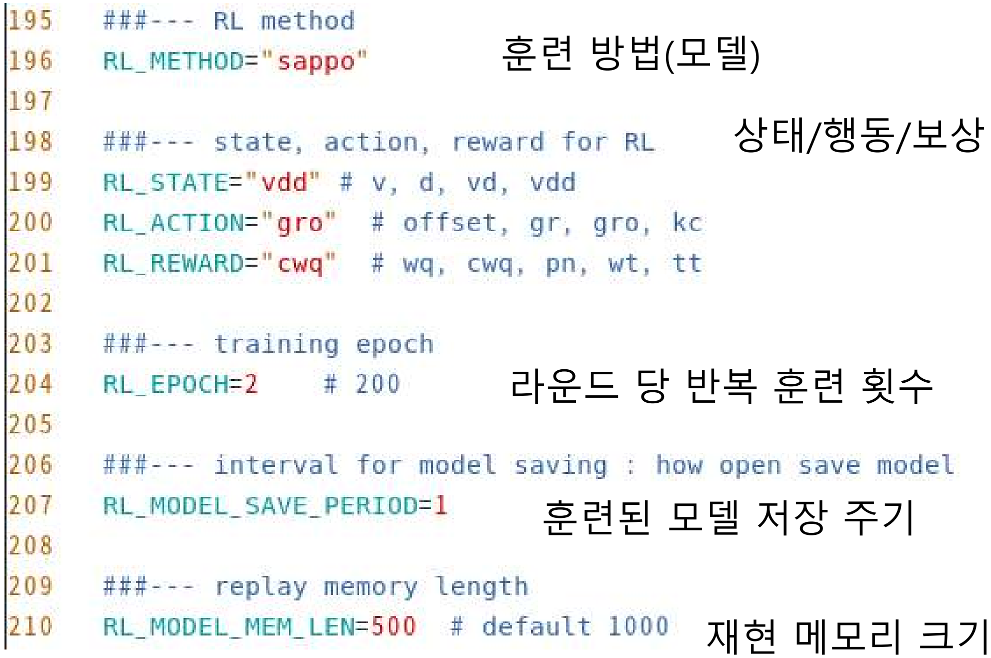
시나리오 상의 map 이름

최적화 대상 SA들;

주의) 콤마(comma)로 구분하며, 중간에 공백이 있어야 함.
SA 101(O)
SA101 (X) : 중간에 공백이 없어서 오동작

스크립트 이용한 실험 설정 : (강화학습 파라미터) 모델, 상태/행동/보상,…

제어 데몬 프로그램 실행 인자



```
###--- RL method
RL_METHOD="sappo"

###--- state, action, reward for RL
RL_STATE="vdd" # v, d, vd, vdd
RL_ACTION="gro"  # offset, gr, gro, kc
RL_REWARD="cwq"  # wq, cwq, pn, wt, tt

###--- training epoch
RL_EPOCH=2     # 200

###--- interval for model saving : how open save model
RL_MODEL_SAVE_PERIOD=1

###--- replay memory length
RL_MODEL_MEM_LEN=500  # default 1000
```

훈련 방법(모델)

상태/행동/보상

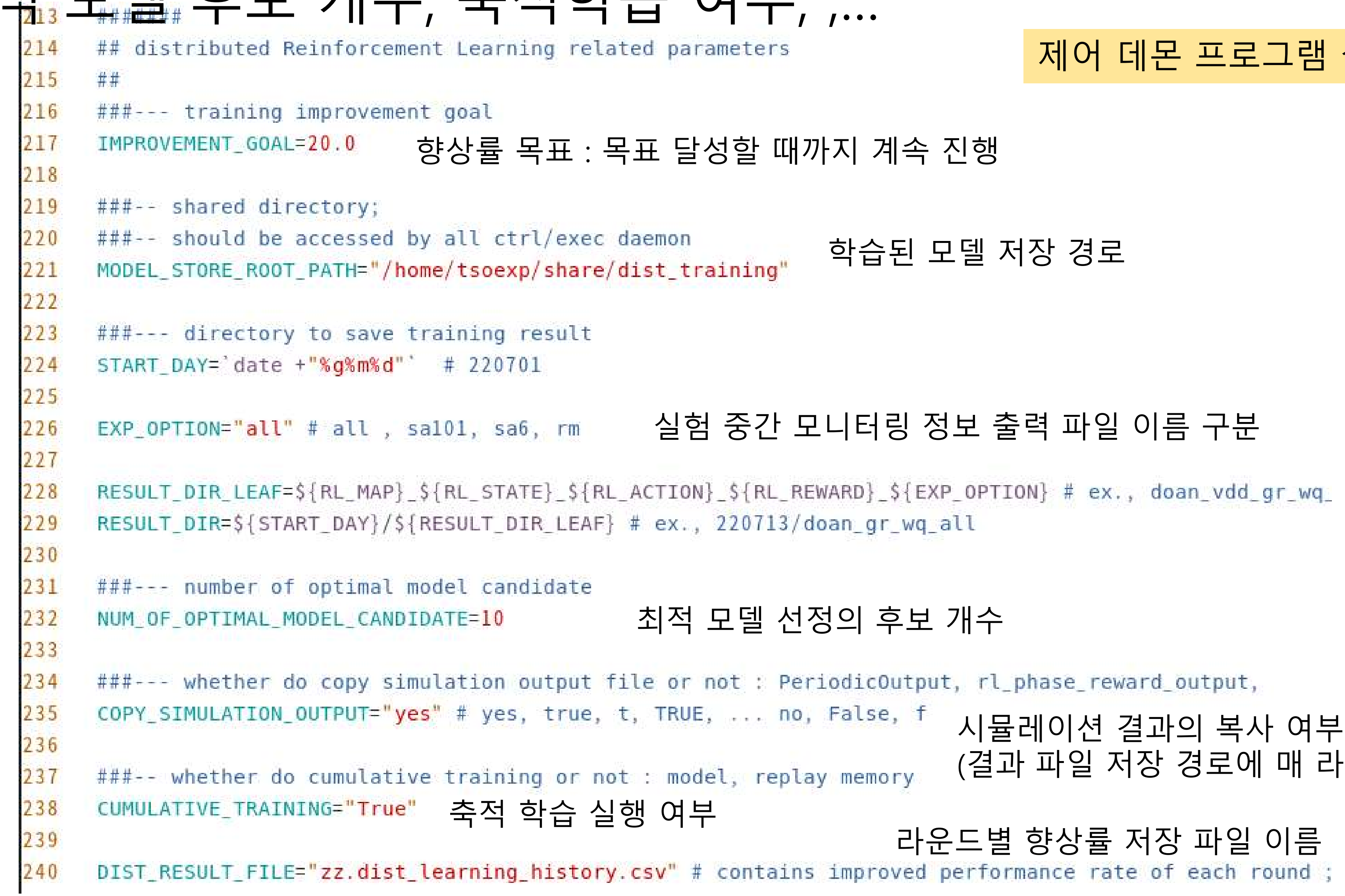
라운드 당 반복 훈련 횟수

훈련된 모델 저장 주기

재현 메모리 크기

스크립트 이용한 실험 설정 : (강화학습 파라미터) 향상률 목표, 저장 경로, 최적 모델 후보 개수, 축적학습 여부, ,…

제어 데몬 프로그램 실행 인자



```
########
## distributed Reinforcement Learning related parameters
##
###--- training improvement goal
IMPROVEMENT_GOAL=20.0
```

향상률 목표 : 목표 달성할 때까지 계속 진행

```

###-- shared directory;
###-- should be accessed by all ctrl/exec daemon
MODEL_STORE_ROOT_PATH="/home/tsoexp/share/dist_training"
```

학습된 모델 저장 경로

```

###--- directory to save training result
START_DAY=`date +"%g%m%d"`  # 220701

EXP_OPTION="all" # all , sa101, sa6, rm
```

실험 중간 모니터링 정보 출력 파일 이름 구분

```

RESULT_DIR_LEAF=${RL_MAP}_${RL_STATE}_${RL_ACTION}_${RL_REWARD}_${EXP_OPTION} # ex., doan_vdd_gr_wq_
RESULT_DIR=${START_DAY}/${RESULT_DIR_LEAF} # ex., 220713/doan_gr_wq_all

###--- number of optimal model candidate
NUM_OF_OPTIMAL_MODEL_CANDIDATE=10
```

최적 모델 선정의 후보 개수

```

###--- whether do copy simulation output file or not : PeriodicOutput, rl_phase_reward_output,
COPY_SIMULATION_OUTPUT="yes" # yes, true, t, TRUE, ... no, False, f
```

시뮬레이션 결과의 복사 여부
(결과 파일 저장 경로에 매 라운드마다 복사)

```

###-- whether do cumulative training or not : model, replay memory
CUMULATIVE_TRAINING="True"
```

축적 학습 실행 여부

라운드별 향상률 저장 파일 이름

```

DIST_RESULT_FILE="zz.dist_learning_history.csv" # contains improved performance rate of each round ;
```

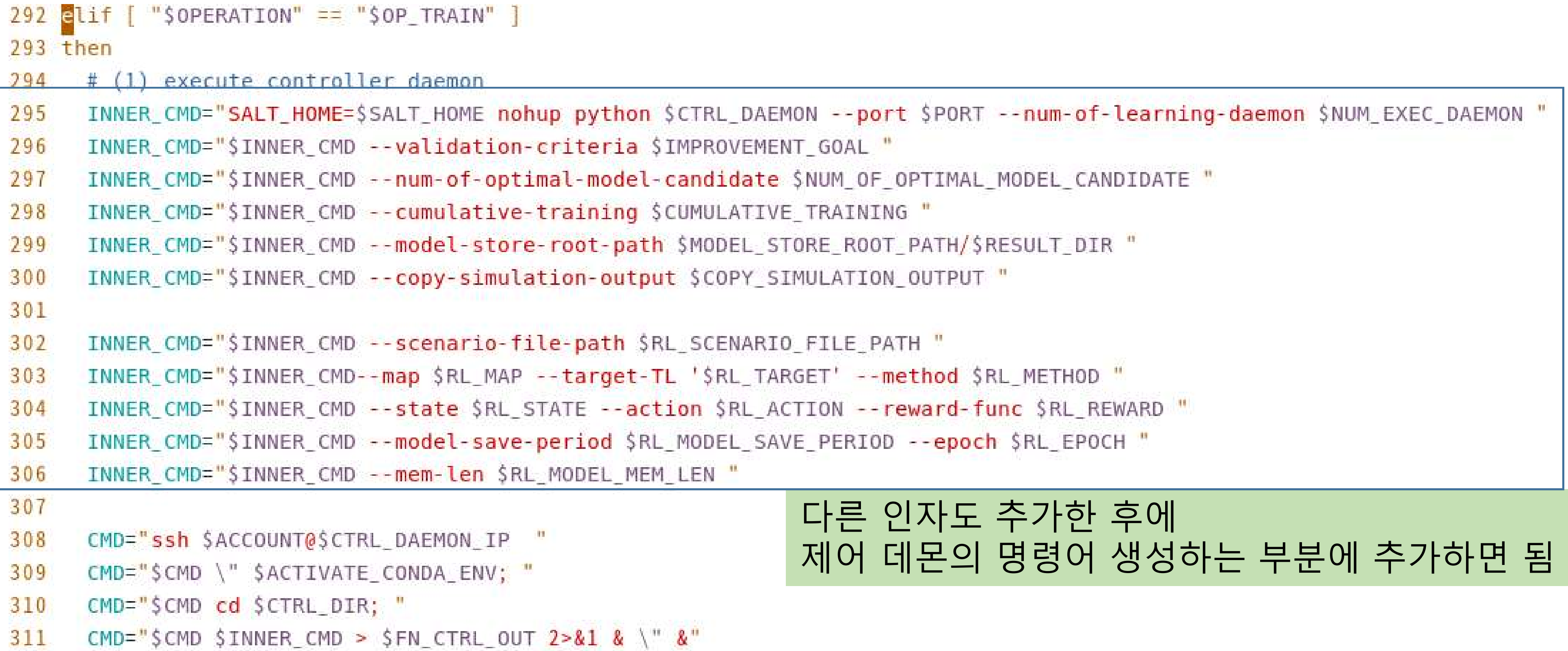
스크립트 이용한 실험 설정 : 다른 인자 추가....

제어 데몬

실행

명령어

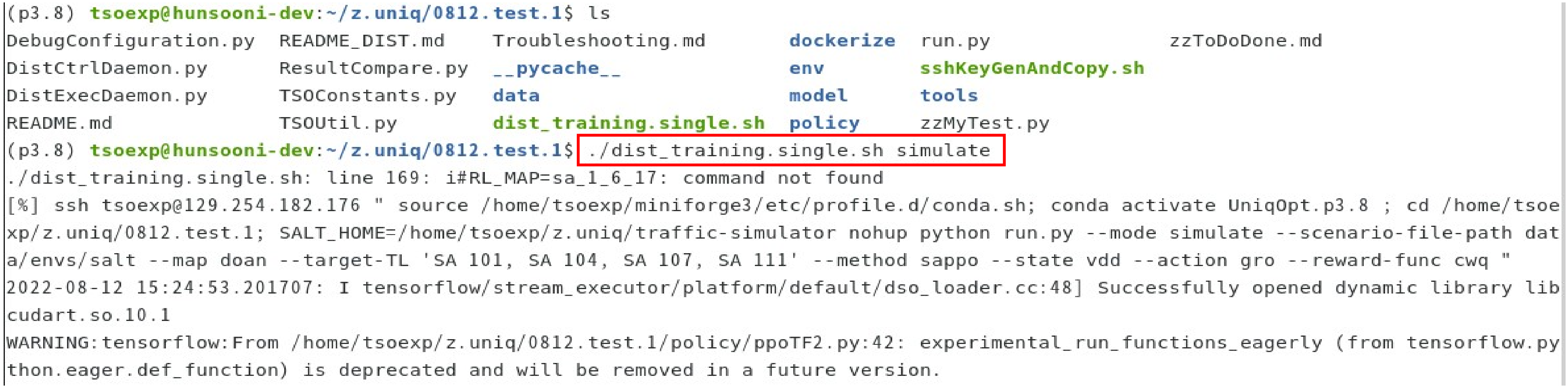
생성 부분



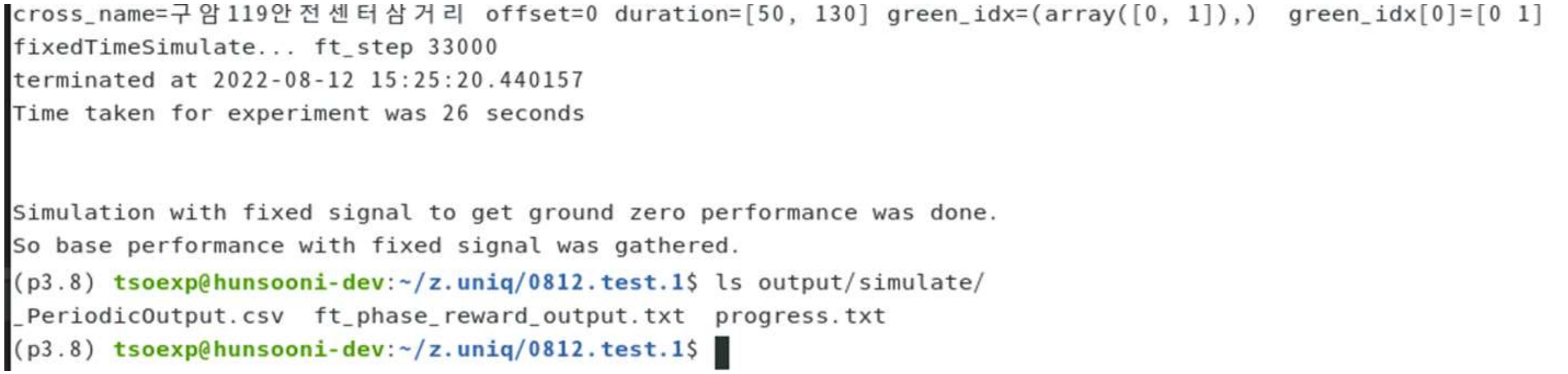
```
elif [ "$OPERATION" == "$OP_TRAIN" ]
then
  # (1) execute controller daemon
  INNER_CMD="SALT_HOME=$SALT_HOME nohup python $CTRL_DAEMON --port $PORT --num-of-learning-daemon $NUM_EXEC_DAEMON "
  INNER_CMD="$INNER_CMD --validation-criteria $IMPROVEMENT_GOAL "
  INNER_CMD="$INNER_CMD --num-of-optimal-model-candidate $NUM_OF_OPTIMAL_MODEL_CANDIDATE "
  INNER_CMD="$INNER_CMD --cumulative-training $CUMULATIVE_TRAINING "
  INNER_CMD="$INNER_CMD --model-store-root-path $MODEL_STORE_ROOT_PATH/$RESULT_DIR "
  INNER_CMD="$INNER_CMD --copy-simulation-output $COPY_SIMULATION_OUTPUT "

  INNER_CMD="$INNER_CMD --scenario-file-path $RL_SCENARIO_FILE_PATH "
  INNER_CMD="$INNER_CMD--map $RL_MAP --target-TL '$RL_TARGET' --method $RL_METHOD "
  INNER_CMD="$INNER_CMD --state $RL_STATE --action $RL_ACTION --reward-func $RL_REWARD "
  INNER_CMD="$INNER_CMD --model-save-period $RL_MODEL_SAVE_PERIOD --epoch $RL_EPOCH "
  INNER_CMD="$INNER_CMD --mem-len $RL_MODEL_MEM_LEN "

  CMD="ssh $ACCOUNT@$CTRL_DAEMON_IP  "
  CMD="$CMD \" $ACTIVATE_CONDA_ENV; "
  CMD="$CMD cd $CTRL_DIR; "
  CMD="$CMD $INNER_CMD > $FN_CTRL_OUT 2>&1 & \" &"
```

다른 인자도 추가한 후에
제어 데몬의 명령어 생성하는 부분에 추가하면 됨

스크립트 이용한 실험 수행 : 고정 신호 실행

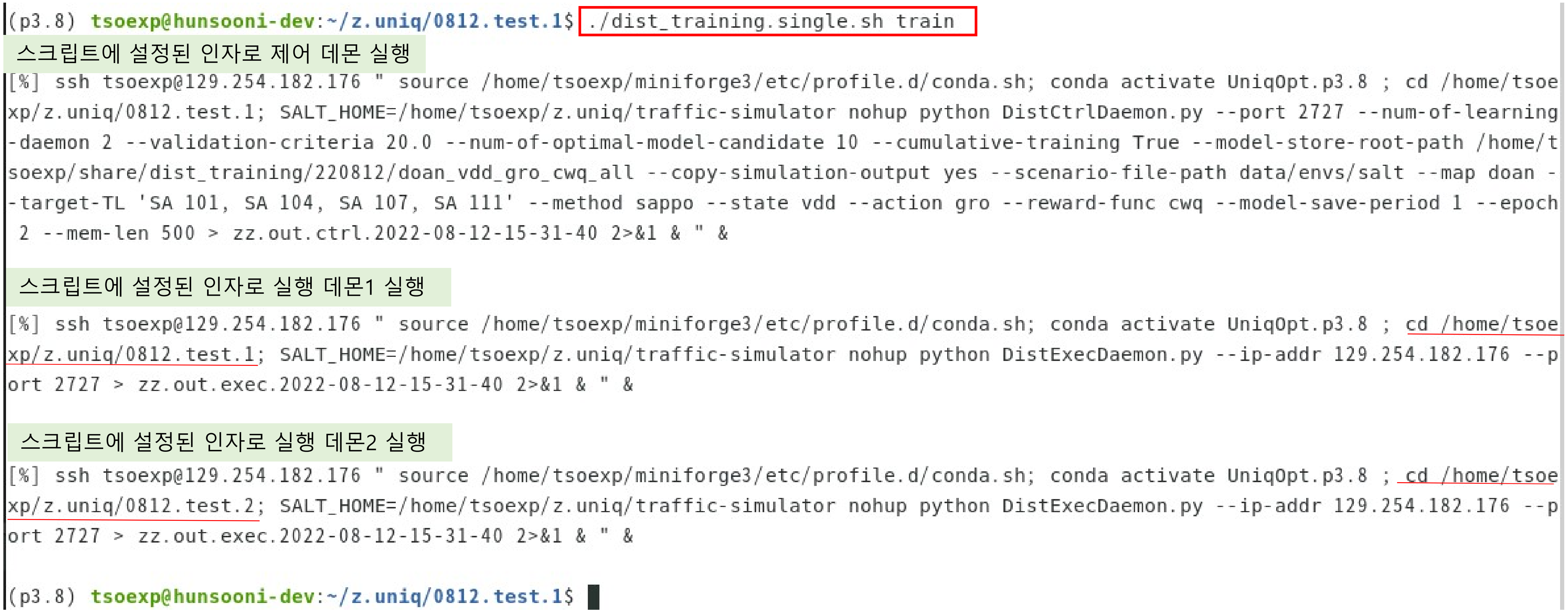


```
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ls
DebugConfiguration.py  README_DIST.md     Troubleshooting.md       dockerize  run.py                 zzToDoDone.md
DistCtrlDaemon.py      ResultCompare.py   __pycache__              env        sshKeyGenAndCopy.sh
DistExecDaemon.py      TSOConstants.py    data                     model      tools
README.md              TSOUtil.py         dist_training.single.sh  policy     zzMyTest.py
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ./dist_training.single.sh simulate
./dist_training.single.sh: line 169: i#RL_MAP=sa_1_6_17: command not found
[%] ssh tsoexp@129.254.182.176 " source /home/tsoexp/miniforge3/etc/profile.d/conda.sh; conda activate UniqOpt.p3.8 ; cd /home/tsoe
xp/z.uniq/0812.test.1; SALT_HOME=/home/tsoexp/z.uniq/traffic-simulator nohup python run.py --mode simulate --scenario-file-path dat
a/envs/salt --map doan --target-TL 'SA 101, SA 104, SA 107, SA 111' --method sappo --state vdd --action gro --reward-func cwq "
2022-08-12 15:24:53.201707: I tensorflow/stream_executor/platform/default/dso_loader.cc:48] Successfully opened dynamic library lib
cudart.so.10.1
WARNING:tensorflow:From /home/tsoexp/z.uniq/0812.test.1/policy/ppoTF2.py:42: experimental_run_functions_eagerly (from tensorflow.py
thon.eager.def_function) is deprecated and will be removed in a future version.
```



```
cross_name=구암119안전센터삼거리  offset=0 duration=[50, 130] green_idx=(array([0, 1]),)  green_idx[0]=[0 1]
fixedTimeSimulate... ft_step 33000
terminated at 2022-08-12 15:25:20.440157
Time taken for experiment was 26 seconds


Simulation with fixed signal to get ground zero performance was done.
So base performance with fixed signal was gathered.
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ls output/simulate/
_PeriodicOutput.csv  ft_phase_reward_output.txt  progress.txt
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ 
```

스크립트 이용한 실험 수행 : 축적 훈련 실행(1/2)



```
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ./dist_training.single.sh train
```

스크립트에 설정된 인자로 제어 데몬 실행

```
[%] ssh tsoexp@129.254.182.176 " source /home/tsoexp/miniforge3/etc/profile.d/conda.sh; conda activate UniqOpt.p3.8 ; cd /home/tsoe
xp/z.uniq/0812.test.1; SALT_HOME=/home/tsoexp/z.uniq/traffic-simulator nohup python DistCtrlDaemon.py --port 2727 --num-of-learning
-daemon 2 --validation-criteria 20.0 --num-of-optimal-model-candidate 10 --cumulative-training True --model-store-root-path /home/t
soexp/share/dist_training/220812/doan_vdd_gro_cwq_all --copy-simulation-output yes --scenario-file-path data/envs/salt --map doan -
-target-TL 'SA 101, SA 104, SA 107, SA 111' --method sappo --state vdd --action gro --reward-func cwq --model-save-period 1 --epoch
 2 --mem-len 500 > zz.out.ctrl.2022-08-12-15-31-40 2>&1 & " &
```

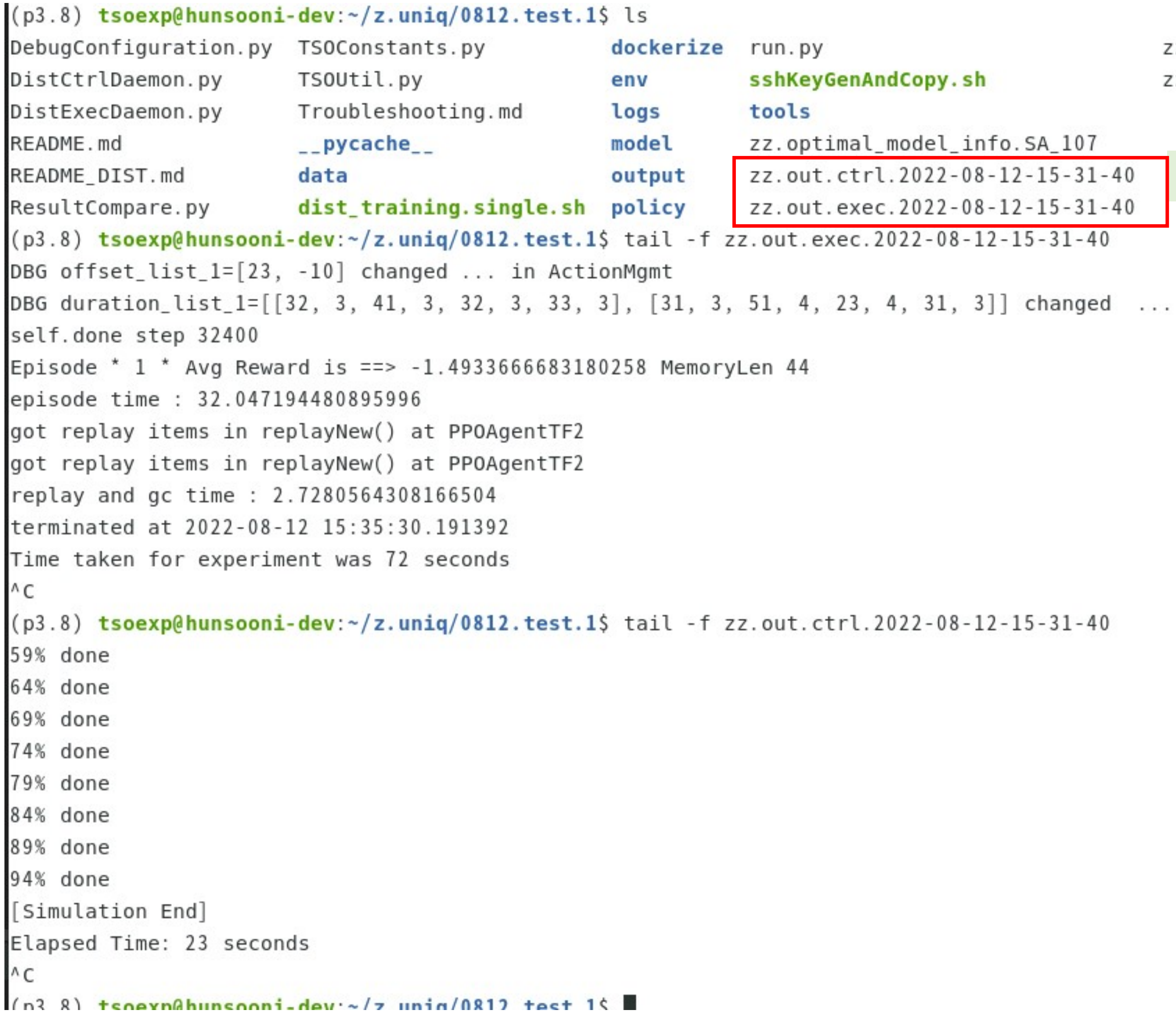
스크립트에 설정된 인자로 실행 데몬1 실행

```
[%] ssh tsoexp@129.254.182.176 " source /home/tsoexp/miniforge3/etc/profile.d/conda.sh; conda activate UniqOpt.p3.8 ; cd /home/tsoe
xp/z.uniq/0812.test.1; SALT_HOME=/home/tsoexp/z.uniq/traffic-simulator nohup python DistExecDaemon.py --ip-addr 129.254.182.176 --p
ort 2727 > zz.out.exec.2022-08-12-15-31-40 2>&1 & " &
```

스크립트에 설정된 인자로 실행 데몬2 실행

```
[%] ssh tsoexp@129.254.182.176 " source /home/tsoexp/miniforge3/etc/profile.d/conda.sh; conda activate UniqOpt.p3.8 ; cd /home/tsoe
xp/z.uniq/0812.test.2; SALT_HOME=/home/tsoexp/z.uniq/traffic-simulator nohup python DistExecDaemon.py --ip-addr 129.254.182.176 --p
ort 2727 > zz.out.exec.2022-08-12-15-31-40 2>&1 & " &

(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ █
```

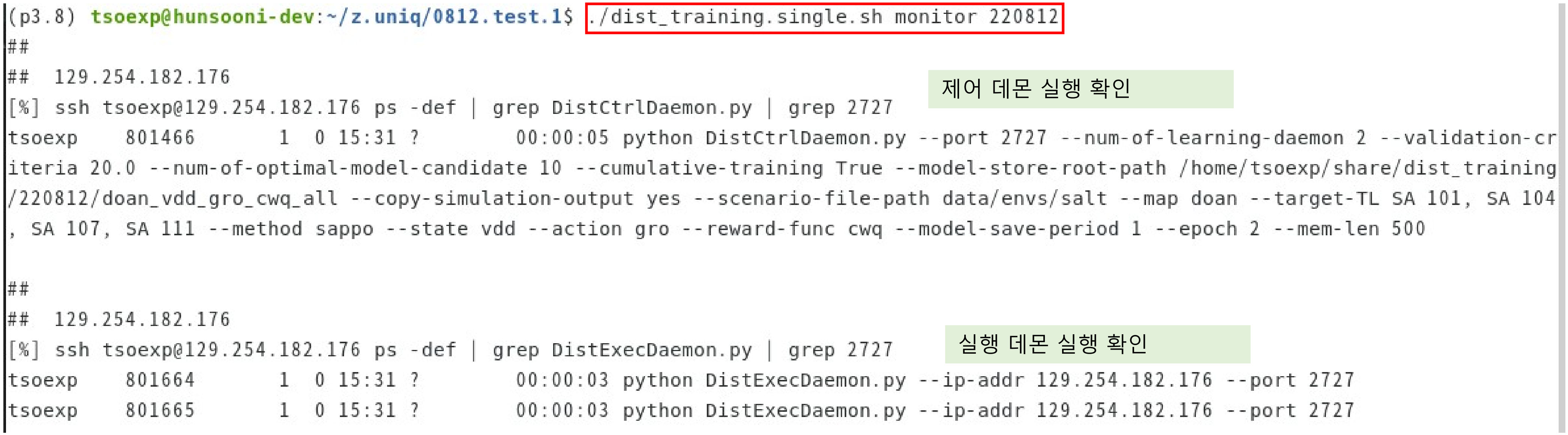
스크립트 이용한 실험 수행 : 축적 훈련 실행(2/2)



```
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ls
DebugConfiguration.py  TSOConstants.py          dockerize  run.py                          zzMyTest.py
DistCtrlDaemon.py      TSOUtil.py               env        sshKeyGenAndCopy.sh             zzToDoDone.md
DistExecDaemon.py      Troubleshooting.md       logs       tools
README.md              __pycache__              model      zz.optimal_model_info.SA_107
README_DIST.md         data                     output     zz.out.ctrl.2022-08-12-15-31-40
ResultCompare.py       dist_training.single.sh  policy     zz.out.exec.2022-08-12-15-31-40
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ tail -f zz.out.exec.2022-08-12-15-31-40
DBG offset_list_1=[23, -10] changed ... in ActionMgmt
DBG duration_list_1=[[32, 3, 41, 3, 32, 3, 33, 3], [31, 3, 51, 4, 23, 4, 31, 3]] changed  ... in ActionMgmt
self.done step 32400
Episode * 1 * Avg Reward is ==> -1.4933666683180258 MemoryLen 44
episode time : 32.047194480895996
got replay items in replayNew() at PPOAgentTF2
got replay items in replayNew() at PPOAgentTF2
replay and gc time : 2.7280564308166504
terminated at 2022-08-12 15:35:30.191392
Time taken for experiment was 72 seconds
^C
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ tail -f zz.out.ctrl.2022-08-12-15-31-40
59% done
64% done
69% done
74% done
79% done
84% done
89% done
94% done
[Simulation End]
Elapsed Time: 23 seconds
^C
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$
```

실험 중간 dump 파일 생성됨

스크립트 이용한 실험 수행 : 실험 모니터링(1/2)



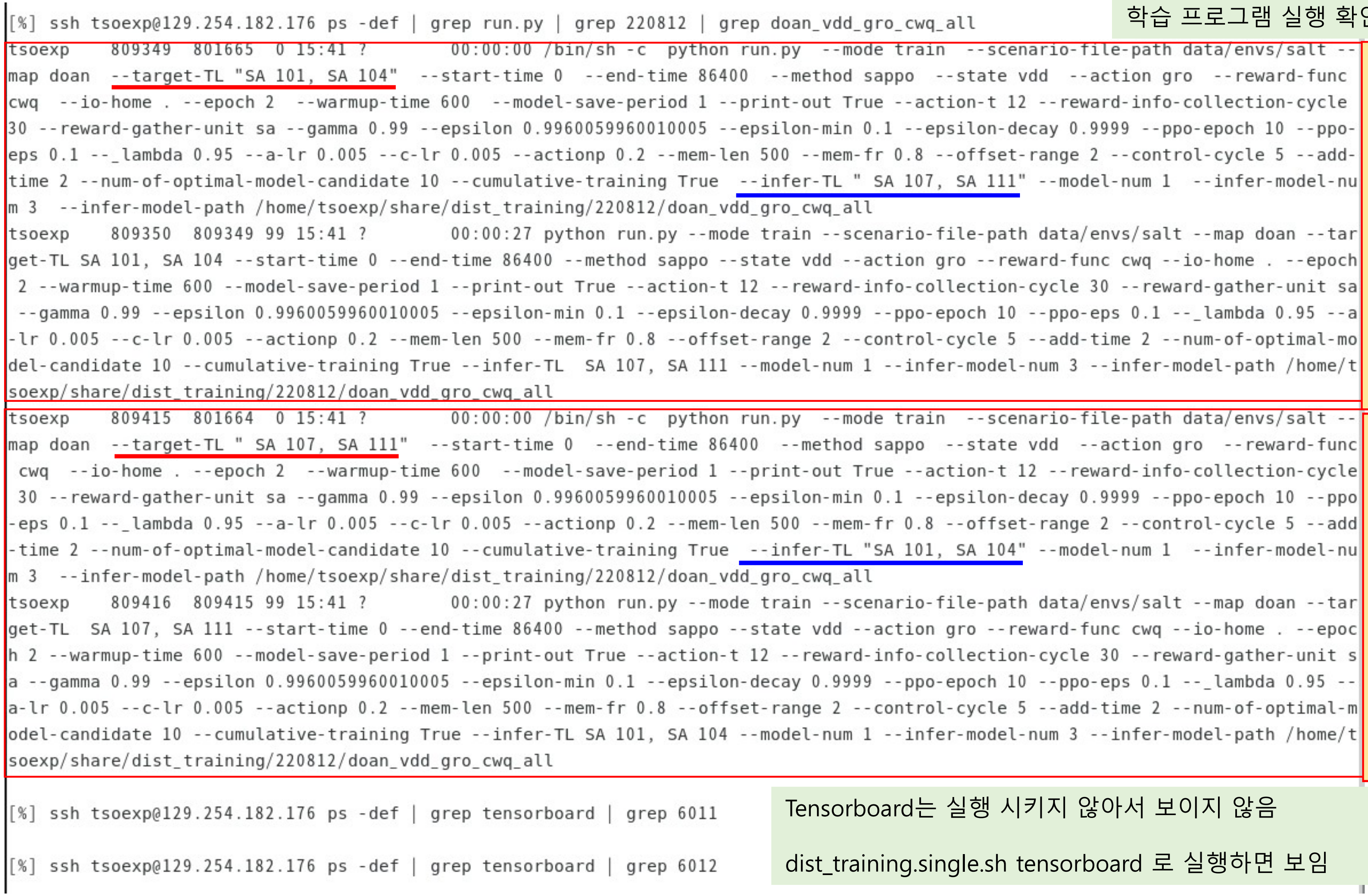
```
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ./dist_training.single.sh monitor 220812
##
##  129.254.182.176
[%] ssh tsoexp@129.254.182.176 ps -def | grep DistCtrlDaemon.py | grep 2727
tsoexp    801466       1  0 15:31 ?        00:00:05 python DistCtrlDaemon.py --port 2727 --num-of-learning-daemon 2 --validation-cr
iteria 20.0 --num-of-optimal-model-candidate 10 --cumulative-training True --model-store-root-path /home/tsoexp/share/dist_training
/220812/doan_vdd_gro_cwq_all --copy-simulation-output yes --scenario-file-path data/envs/salt --map doan --target-TL SA 101, SA 104
, SA 107, SA 111 --method sappo --state vdd --action gro --reward-func cwq --model-save-period 1 --epoch 2 --mem-len 500

##
##  129.254.182.176
[%] ssh tsoexp@129.254.182.176 ps -def | grep DistExecDaemon.py | grep 2727
tsoexp    801664       1  0 15:31 ?        00:00:03 python DistExecDaemon.py --ip-addr 129.254.182.176 --port 2727
tsoexp    801665       1  0 15:31 ?        00:00:03 python DistExecDaemon.py --ip-addr 129.254.182.176 --port 2727
```

제어 데몬 실행 확인

실행 데몬 실행 확인

스크립트 이용한 실험 수행 : 실험 모니터링(2/2)

학습 프로그램 실행 확인



```
[%] ssh tsoexp@129.254.182.176 ps -def | grep run.py | grep 220812 | grep doan_vdd_gro_cwq_all
tsoexp    809349  801665  0 15:41 ?        00:00:00 /bin/sh -c  python run.py  --mode train  --scenario-file-path data/envs/salt --
map doan  --target-TL "SA 101, SA 104"  --start-time 0  --end-time 86400  --method sappo  --state vdd  --action gro  --reward-func
cwq  --io-home . --epoch 2  --warmup-time 600  --model-save-period 1 --print-out True --action-t 12 --reward-info-collection-cycle
30 --reward-gather-unit sa --gamma 0.99 --epsilon 0.9960059960010005 --epsilon-min 0.1 --epsilon-decay 0.9999 --ppo-epoch 10 --ppo-
eps 0.1 --_lambda 0.95 --a-lr 0.005 --c-lr 0.005 --actionp 0.2 --mem-len 500 --mem-fr 0.8 --offset-range 2 --control-cycle 5 --add-
time 2 --num-of-optimal-model-candidate 10 --cumulative-training True  --infer-TL " SA 107, SA 111" --model-num 1  --infer-model-nu
m 3  --infer-model-path /home/tsoexp/share/dist_training/220812/doan_vdd_gro_cwq_all
tsoexp    809350  809349 99 15:41 ?        00:00:27 python run.py --mode train --scenario-file-path data/envs/salt --map doan --tar
get-TL SA 101, SA 104 --start-time 0 --end-time 86400 --method sappo --state vdd --action gro --reward-func cwq --io-home . --epoch
 2 --warmup-time 600 --model-save-period 1 --print-out True --action-t 12 --reward-info-collection-cycle 30 --reward-gather-unit sa
 --gamma 0.99 --epsilon 0.9960059960010005 --epsilon-min 0.1 --epsilon-decay 0.9999 --ppo-epoch 10 --ppo-eps 0.1 --_lambda 0.95 --a
-lr 0.005 --c-lr 0.005 --actionp 0.2 --mem-len 500 --mem-fr 0.8 --offset-range 2 --control-cycle 5 --add-time 2 --num-of-optimal-mo
del-candidate 10 --cumulative-training True --infer-TL  SA 107, SA 111 --model-num 1 --infer-model-num 3 --infer-model-path /home/t
soexp/share/dist_training/220812/doan_vdd_gro_cwq_all
tsoexp    809415  801664  0 15:41 ?        00:00:00 /bin/sh -c  python run.py  --mode train  --scenario-file-path data/envs/salt --
map doan  --target-TL " SA 107, SA 111"  --start-time 0  --end-time 86400  --method sappo  --state vdd  --action gro  --reward-func
 cwq  --io-home . --epoch 2  --warmup-time 600  --model-save-period 1 --print-out True --action-t 12 --reward-info-collection-cycle
 30 --reward-gather-unit sa --gamma 0.99 --epsilon 0.9960059960010005 --epsilon-min 0.1 --epsilon-decay 0.9999 --ppo-epoch 10 --ppo
-eps 0.1 --_lambda 0.95 --a-lr 0.005 --c-lr 0.005 --actionp 0.2 --mem-len 500 --mem-fr 0.8 --offset-range 2 --control-cycle 5 --add
-time 2 --num-of-optimal-model-candidate 10 --cumulative-training True  --infer-TL "SA 101, SA 104" --model-num 1  --infer-model-nu
m 3  --infer-model-path /home/tsoexp/share/dist_training/220812/doan_vdd_gro_cwq_all
tsoexp    809416  809415 99 15:41 ?        00:00:27 python run.py --mode train --scenario-file-path data/envs/salt --map doan --tar
get-TL  SA 107, SA 111 --start-time 0 --end-time 86400 --method sappo --state vdd --action gro --reward-func cwq --io-home . --epoc
h 2 --warmup-time 600 --model-save-period 1 --print-out True --action-t 12 --reward-info-collection-cycle 30 --reward-gather-unit s
a --gamma 0.99 --epsilon 0.9960059960010005 --epsilon-min 0.1 --epsilon-decay 0.9999 --ppo-epoch 10 --ppo-eps 0.1 --_lambda 0.95 --
a-lr 0.005 --c-lr 0.005 --actionp 0.2 --mem-len 500 --mem-fr 0.8 --offset-range 2 --control-cycle 5 --add-time 2 --num-of-optimal-m
odel-candidate 10 --cumulative-training True --infer-TL SA 101, SA 104 --model-num 1 --infer-model-num 3 --infer-model-path /home/t
soexp/share/dist_training/220812/doan_vdd_gro_cwq_all

[%] ssh tsoexp@129.254.182.176 ps -def | grep tensorboard | grep 6011

[%] ssh tsoexp@129.254.182.176 ps -def | grep tensorboard | grep 6012
```

SA 101,
SA 104
학습

SA 107,
SA 111
추론

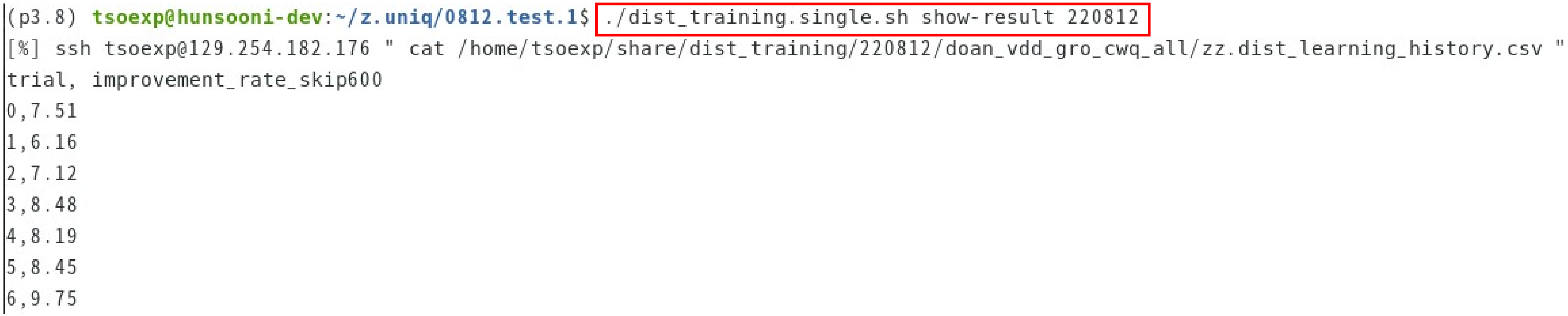
SA 107,
SA 111
학습

SA 101,
SA 104
추론

Tensorboard는 실행 시키지 않아서 보이지 않음

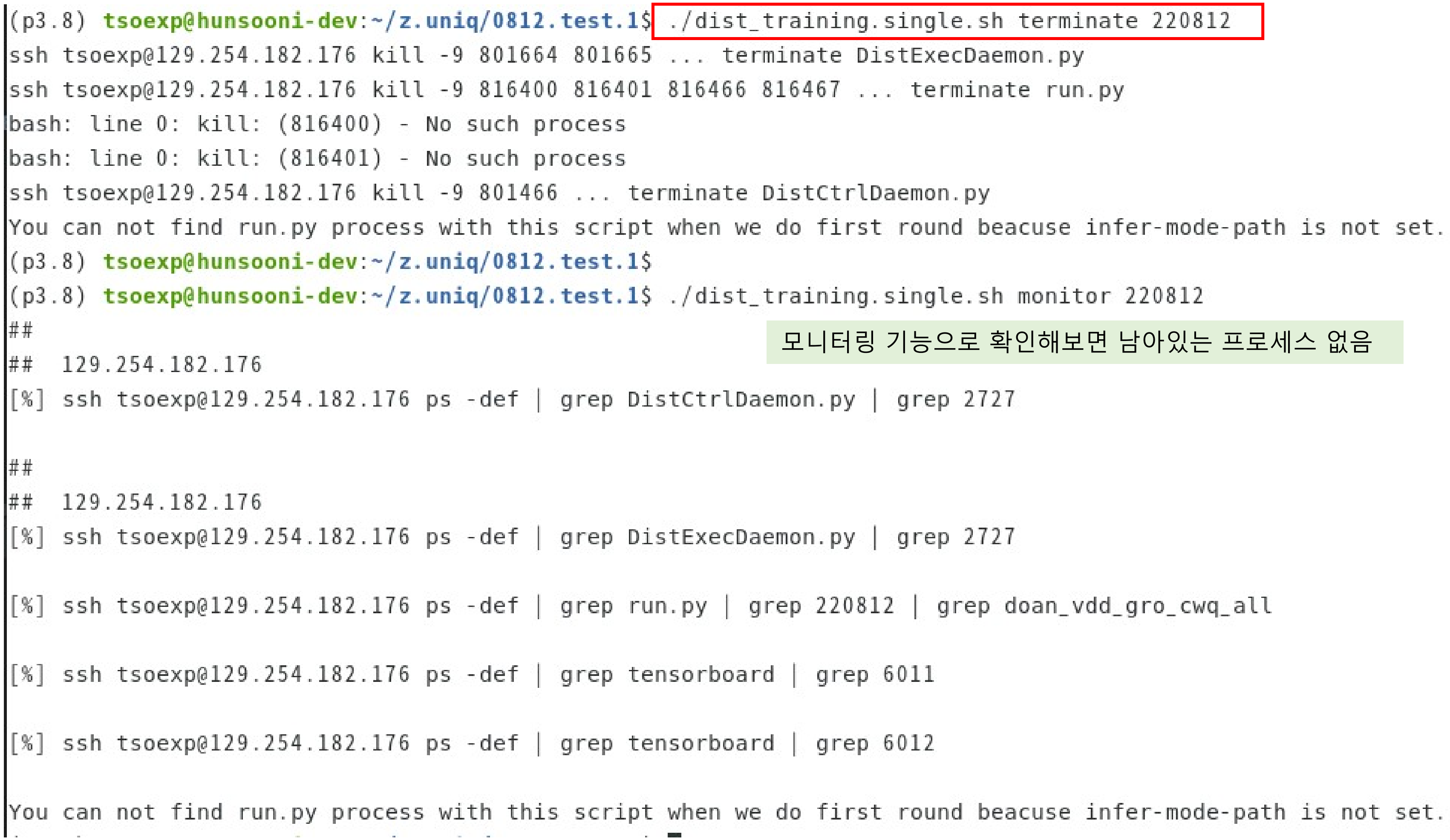
dist_training.single.sh tensorboard 로 실행하면 보임

스크립트 이용한 실험 수행 : 결과 모니터링



```
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ./dist_training.single.sh show-result 220812
[%] ssh tsoexp@129.254.182.176 " cat /home/tsoexp/share/dist_training/220812/doan_vdd_gro_cwq_all/zz.dist_learning_history.csv "
trial, improvement_rate_skip600
0,7.51
1,6.16
2,7.12
3,8.48
4,8.19
5,8.45
6,9.75
```

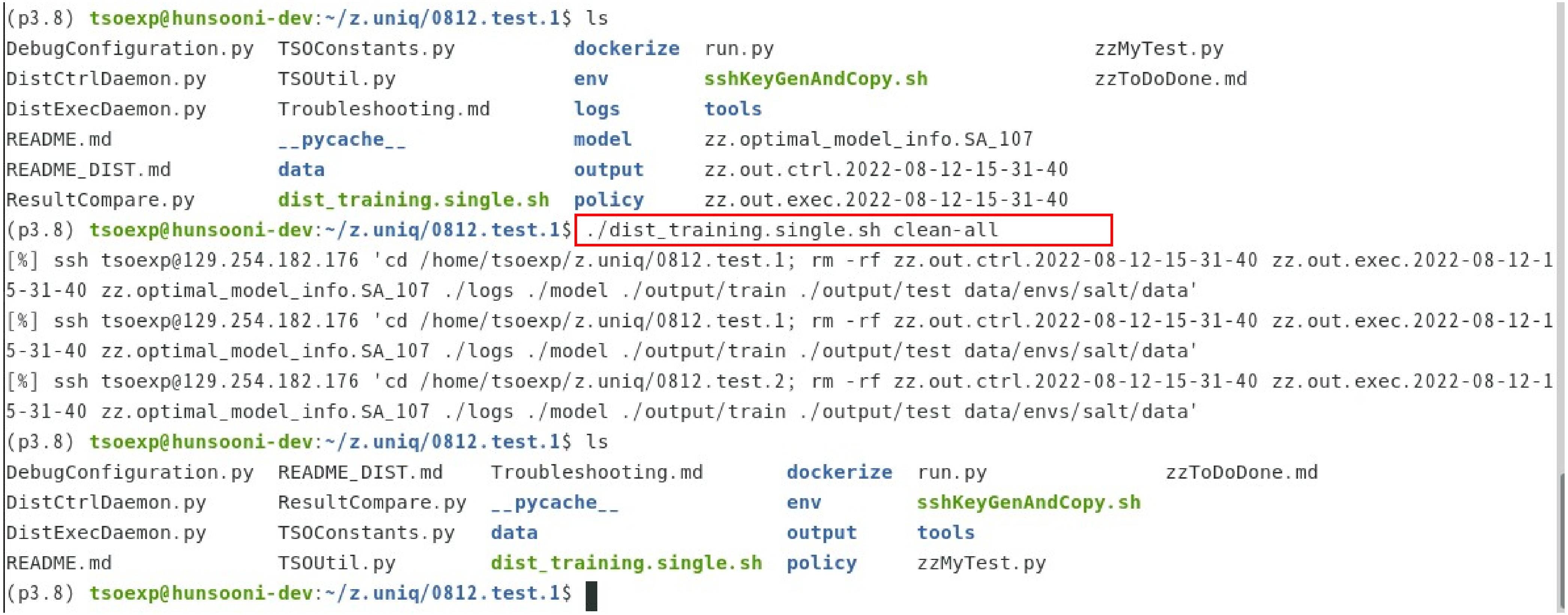
스크립트 이용한 실험 수행 : 실험 강제 종료



```
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ./dist_training.single.sh terminate 220812
ssh tsoexp@129.254.182.176 kill -9 801664 801665 ... terminate DistExecDaemon.py
ssh tsoexp@129.254.182.176 kill -9 816400 816401 816466 816467 ... terminate run.py
bash: line 0: kill: (816400) - No such process
bash: line 0: kill: (816401) - No such process
ssh tsoexp@129.254.182.176 kill -9 801466 ... terminate DistCtrlDaemon.py
You can not find run.py process with this script when we do first round beacuse infer-mode-path is not set.
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ./dist_training.single.sh monitor 220812
##
##  129.254.182.176
[%] ssh tsoexp@129.254.182.176 ps -def | grep DistCtrlDaemon.py | grep 2727

##
##  129.254.182.176
[%] ssh tsoexp@129.254.182.176 ps -def | grep DistExecDaemon.py | grep 2727

[%] ssh tsoexp@129.254.182.176 ps -def | grep run.py | grep 220812 | grep doan_vdd_gro_cwq_all

[%] ssh tsoexp@129.254.182.176 ps -def | grep tensorboard | grep 6011

[%] ssh tsoexp@129.254.182.176 ps -def | grep tensorboard | grep 6012

You can not find run.py process with this script when we do first round beacuse infer-mode-path is not set.
```

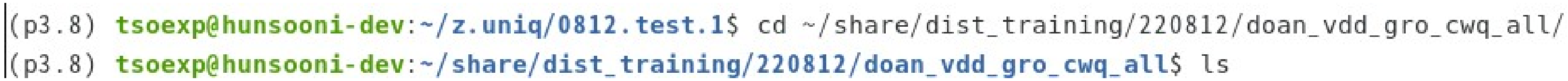
모니터링 기능으로 확인해보면 남아있는 프로세스 없음

스크립트 이용한 실험 수행 : 실험 중간 dump 파일 정리



```
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ls
DebugConfiguration.py  TSOConstants.py          dockerize  run.py                           zzMyTest.py
DistCtrlDaemon.py      TSOUtil.py               env        sshKeyGenAndCopy.sh              zzToDoDone.md
DistExecDaemon.py      Troubleshooting.md       logs       tools
README.md              __pycache__              model      zz.optimal_model_info.SA_107
README_DIST.md         data                     output     zz.out.ctrl.2022-08-12-15-31-40
ResultCompare.py       dist_training.single.sh  policy     zz.out.exec.2022-08-12-15-31-40
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ./dist_training.single.sh clean-all
[%] ssh tsoexp@129.254.182.176 'cd /home/tsoexp/z.uniq/0812.test.1; rm -rf zz.out.ctrl.2022-08-12-15-31-40 zz.out.exec.2022-08-12-1
5-31-40 zz.optimal_model_info.SA_107 ./logs ./model ./output/train ./output/test data/envs/salt/data'
[%] ssh tsoexp@129.254.182.176 'cd /home/tsoexp/z.uniq/0812.test.1; rm -rf zz.out.ctrl.2022-08-12-15-31-40 zz.out.exec.2022-08-12-1
5-31-40 zz.optimal_model_info.SA_107 ./logs ./model ./output/train ./output/test data/envs/salt/data'
[%] ssh tsoexp@129.254.182.176 'cd /home/tsoexp/z.uniq/0812.test.2; rm -rf zz.out.ctrl.2022-08-12-15-31-40 zz.out.exec.2022-08-12-1
5-31-40 zz.optimal_model_info.SA_107 ./logs ./model ./output/train ./output/test data/envs/salt/data'
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ ls
DebugConfiguration.py  README_DIST.md    Troubleshooting.md       dockerize  run.py               zzToDoDone.md
DistCtrlDaemon.py      ResultCompare.py  __pycache__              env        sshKeyGenAndCopy.sh
DistExecDaemon.py      TSOConstants.py   data                     output     tools
README.md              TSOUtil.py        dist_training.single.sh  policy     zzMyTest.py
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$
```

스크립트 이용한 실험 수행 : 결과 파일/모델 저장

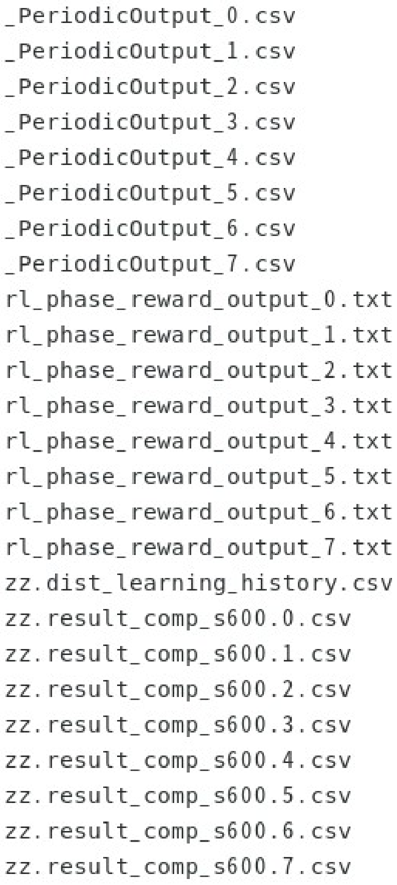


```
(p3.8) tsoexp@hunsooni-dev:~/z.uniq/0812.test.1$ cd ~/share/dist_training/220812/doan_vdd_gro_cwq_all/
(p3.8) tsoexp@hunsooni-dev:~/share/dist_training/220812/doan_vdd_gro_cwq_all$ ls
```

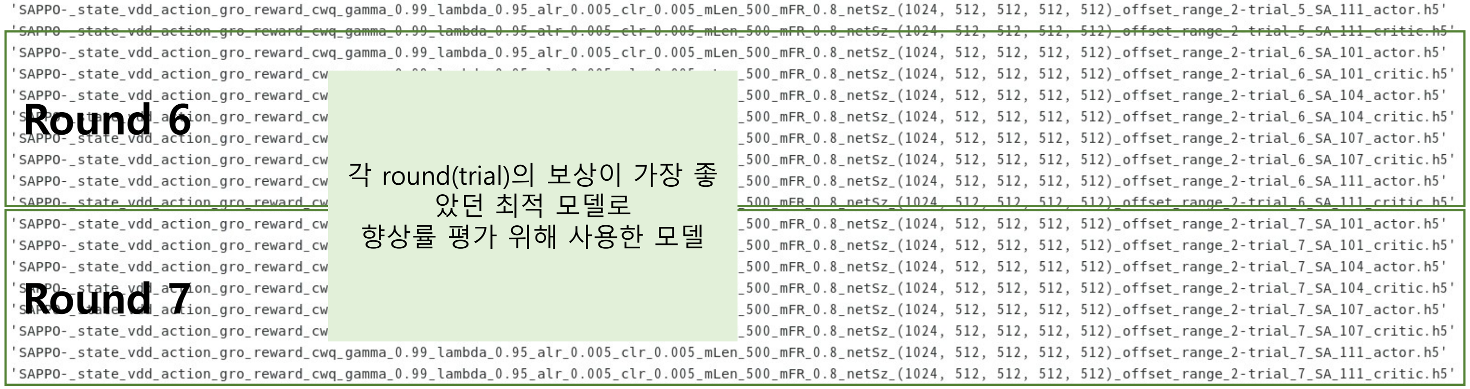
시뮬레이터 결과 파일

최적화 결과 파일

고정 신호 제어와 강화학습 제어의 결과 비교 파일



```
_PeriodicOutput_0.csv
_PeriodicOutput_1.csv
_PeriodicOutput_2.csv
_PeriodicOutput_3.csv
_PeriodicOutput_4.csv
_PeriodicOutput_5.csv
_PeriodicOutput_6.csv
_PeriodicOutput_7.csv
rl_phase_reward_output_0.txt
rl_phase_reward_output_1.txt
rl_phase_reward_output_2.txt
rl_phase_reward_output_3.txt
rl_phase_reward_output_4.txt
rl_phase_reward_output_5.txt
rl_phase_reward_output_6.txt
rl_phase_reward_output_7.txt
zz.dist_learning_history.csv
zz.result_comp_s600.0.csv
zz.result_comp_s600.1.csv
zz.result_comp_s600.2.csv
zz.result_comp_s600.3.csv
zz.result_comp_s600.4.csv
zz.result_comp_s600.5.csv
zz.result_comp_s600.6.csv
zz.result_comp_s600.7.csv
```



```
'SAPPO-_state_vdd_action_gro_reward_cwq_gamma_0.99_lambda_0.95_alr_0.005_clr_0.005_mLen_500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_5_SA_111_actor.h5'
'SAPPO-_state_vdd_action_gro_reward_cwq_gamma_0.99_lambda_0.95_alr_0.005_clr_0.005_mLen_500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_5_SA_111_critic.h5'
'SAPPO-_state_vdd_action_gro_reward_cwq_gamma_0.99_lambda_0.95_alr_0.005_clr_0.005_mLen_500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_6_SA_101_actor.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_6_SA_101_critic.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_6_SA_104_actor.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_6_SA_104_critic.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_6_SA_107_actor.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_6_SA_107_critic.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_6_SA_111_actor.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_6_SA_111_critic.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_7_SA_101_actor.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_7_SA_101_critic.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_7_SA_104_actor.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_7_SA_104_critic.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_7_SA_107_actor.h5'
'SAPPO-_state_vdd_action_gro_reward_cw                                    _500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_7_SA_107_critic.h5'
'SAPPO-_state_vdd_action_gro_reward_cwq_gamma_0.99_lambda_0.95_alr_0.005_clr_0.005_mLen_500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_7_SA_111_actor.h5'
'SAPPO-_state_vdd_action_gro_reward_cwq_gamma_0.99_lambda_0.95_alr_0.005_clr_0.005_mLen_500_mFR_0.8_netSz_(1024, 512, 512, 512, 512)_offset_range_2-trial_7_SA_111_critic.h5'
```

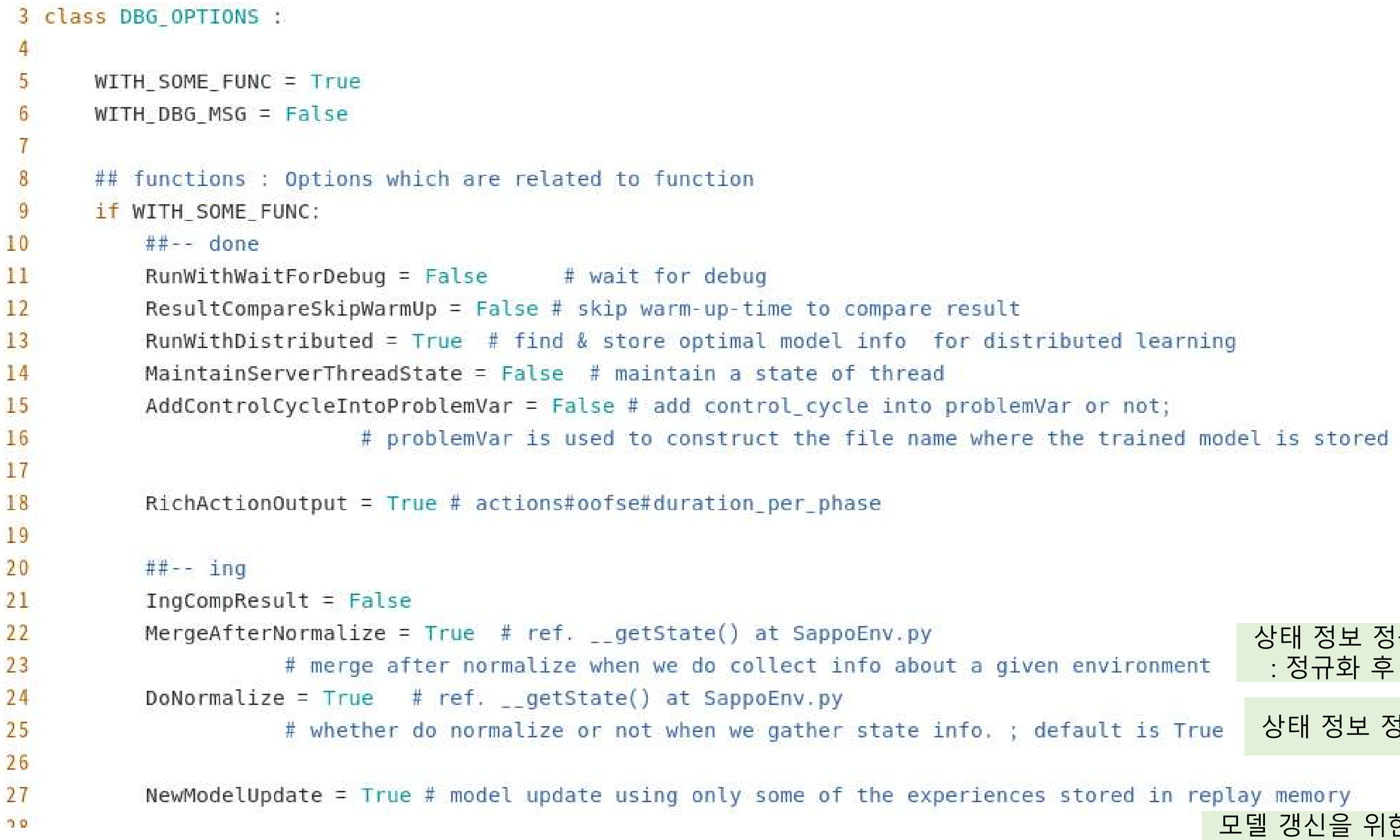
Round 6

Round 7

각 round(trial)의 보상이 가장 좋았던 최적 모델로
향상률 평가 위해 사용한 모델

Bonus…. 실험 기능 추가/제외

- DebugConfiguration.py에 옵션들 변경



```
class DBG_OPTIONS :

    WITH_SOME_FUNC = True
    WITH_DBG_MSG = False

    ## functions : Options which are related to function
    if WITH_SOME_FUNC:
        ##-- done
        RunWithWaitForDebug = False      # wait for debug
        ResultCompareSkipWarmUp = False # skip warm-up-time to compare result
        RunWithDistributed = True  # find & store optimal model info  for distributed learning
        MaintainServerThreadState = False  # maintain a state of thread
        AddControlCycleIntoProblemVar = False # add control_cycle into problemVar or not;
                         # problemVar is used to construct the file name where the trained model is stored

        RichActionOutput = True # actions#oofse#duration_per_phase

        ##-- ing
        IngCompResult = False
        MergeAfterNormalize = True  # ref. __getState() at SappoEnv.py
                   # merge after normalize when we do collect info about a given environment
        DoNormalize = True   # ref. __getState() at SappoEnv.py
                   # whether do normalize or not when we gather state info. ; default is True

        NewModelUpdate = True # model update using only some of the experiences stored in replay memory
```

상태 정보 정규화 시점 설정
: 정규화 후 상태 정보 병합 혹은 상태 정보 병합 후 정규화

상태 정보 정규화 수행 여부

모델 갱신을 위한 학습시 경험 샘플링 여부
: True이면 sampling, False 이면 모든 경험 이용