

## 471 A Marginal Distributions and Returns

472 We expand the marginal transition distribution ( $\rho_{sim}$ ) definition to be more explicit below.

$$\rho_{sim,t}(s, a, s') := \rho_{sim,t}(s) \pi(a|s) P_{sim}(s'|s, a) \quad (7)$$

$$\rho_{sim,t}(s') := \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} \rho_{sim,t-1}(s, a, s') \quad (8)$$

$$\rho_{sim}(s, a, s') := (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t \rho_{sim,t}(s, a, s') \quad (9)$$

473 where  $\rho_{sim,0}(s) = \rho_0(s)$  is the starting state distribution. Written in a single equation:

$$\rho_{sim}(s, a, s') = (1 - \gamma) \sum_{s_0 \in \mathcal{S}} \rho_0(s_0) \sum_{t=0}^{\infty} \gamma^t \sum_{a_t \in \mathcal{A}} \sum_{s_{t+1} \in \mathcal{S}} \pi(a_t|s_t) P(s_{t+1}|s_t, a_t)$$

474 The expected return can be written more explicitly to show the dependence on the transition function.

475 It then makes the connection to [7](#) more explicit.

$$\begin{aligned} \mathbb{E}_{\pi, P} [G_0] &= \mathbb{E}_{\pi, P} \left[ \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t+1}) \right] \\ &= \sum_{s_0 \in \mathcal{S}} \rho_0(s_0) \sum_{t=0}^{\infty} \gamma^t \sum_{a_t \in \mathcal{A}} \sum_{s_{t+1} \in \mathcal{S}} \pi(a_t|s_t) P(s_{t+1}|s_t, a_t) R(s_t, a_t, s_{t+1}) \end{aligned}$$

476 In the grounded simulator, the action transformer policy  $\pi_g$  transforms the transition function as  
 477 specified in Section [2.2](#). Ideally, such a  $\pi_g \in \Pi_g$  exists. We denote the marginal transition  
 478 distributions in sim and real by  $\rho_{sim}$  and  $\rho_{real}$  respectively, and  $\rho_g \in \mathcal{P}_g$  for the grounded simulator.  
 479 The distribution  $\rho_g$  relies on  $\pi_g \in \Pi_g$  as follows:

$$\rho_g(s, a, s') = (1 - \gamma) \pi(a|s) \sum_{\tilde{a} \in \mathcal{A}} P_{sim}(s'|s, \tilde{a}) \pi_g(\tilde{a}|s, a) \sum_{t=0}^{\infty} \gamma^t p(s_t = s | \pi, P_g) \quad (10)$$

480 The marginal transition distribution of the simulator after action transformation,  $\rho_g(s, a, s')$ , differs  
 481 in Equation [7](#) as follows:

$$\rho_{g,t}(s, a, s') := \rho_{g,t}(s) \pi(a|s) \sum_{\tilde{a} \in \mathcal{A}} \pi_g(\tilde{a}|s, a) P_g(s'|s, \tilde{a}) \quad (11)$$

## 482 B Proofs

### 483 B.1 Proof of Proposition [4.1](#)

484 **Proposition 4.1.** For a given  $\rho_g$  generated by a fixed policy  $\pi$ ,  $P_g$  is the only transition function  
 485 whose marginal transition distribution is  $\rho_g$ .

486 *Proof.* We prove the above statement by contradiction. Consider two transition functions  $P_1$  and  $P_2$   
 487 that have the same marginal distribution  $\rho_\pi$  under the same policy  $\pi$ , but differ in their likelihood for  
 488 at least one transition  $(s, a, s')$ .

$$P_1(s'|s, a) \neq P_2(s'|s, a) \quad (12)$$

489 Let us denote the marginal distributions for  $P_1$  and  $P_2$  under policy  $\pi$  as  $\rho_1^\pi$  and  $\rho_2^\pi$ . Thus,  $\rho_1^\pi(s) =$   
 490  $\rho_2^\pi(s) \forall s \in \mathcal{S}$  and  $\rho_1^\pi(s, a, s') = \rho_2^\pi(s, a, s') \forall s, s' \in \mathcal{S}, a \in \mathcal{A}$ .

491 The marginal likelihood of the above transition for both  $P_1$  and  $P_2$  is:

$$\begin{aligned}\rho_1^\pi(s, a, s') &= \sum_{t=0}^{T-1} \rho_1^\pi(s) \pi(a|s) P_1(s'|s, a) \\ \rho_2^\pi(s, a, s') &= \sum_{t=0}^{T-1} \rho_2^\pi(s) \pi(a|s) P_2(s'|s, a)\end{aligned}$$

492 Since the marginal distributions match, and the policy is the same, this leads to the equality:

$$P_1(s'|s, a) = P_2(s'|s, a) \forall s, s' \in \mathcal{S}, a \in \mathcal{A} \quad (13)$$

493 Equation [13](#) contradicts Equation [12](#), proving our claim.  $\square$

## 494 B.2 Proof of Proposition [4.2](#)

495 **Proposition 4.2.** *If  $P_{real} = P_g$ , then  $\operatorname{argmax}_{\pi \in \Pi} \mathbb{E}_{\pi, P_g}[G_0] = \operatorname{argmax}_{\pi \in \Pi} \mathbb{E}_{\pi, P_{real}}[G_0]$ .*

496 *Proof.* We overload the notation slightly and refer to  $\rho_{real}^\pi$  as the marginal transition distribution  
497 in the real world while following agent policy  $\pi$ . Proposition [4.1](#) still holds under this expanded  
498 notation.

499 From Proposition [4.1](#) if  $P_{real} = P_g$ , we can say that  $\rho_{real}^\pi = \rho_g^\pi \forall \pi \in \Pi$ . From Equation [1](#),  
500  $\mathbb{E}_{\pi, g}[G_0] = \mathbb{E}_{\pi, real}[G_0] \forall \pi \in \Pi$ , and  $\operatorname{argmax}_{\pi \in \Pi} \mathbb{E}_{\pi, g}[G_0] = \operatorname{argmax}_{\pi \in \Pi} \mathbb{E}_{\pi, real}[G_0]$ .  $\square$

## 501 B.3 Proof of Lemma [4.1](#)

502 **Lemma 4.1.**  *$\overline{\text{RL}} \circ \overline{\text{ATIRL}}_\psi(P_{real})$  outputs a marginal transition distribution  $\bar{\rho}_g$  which is equal to  $\tilde{\rho}_g$   
503 induced by  $\text{RL} \circ \text{ATIRL}_\psi(P_{real})$ .*

504 *Proof.* For every  $\rho_g \in \mathcal{P}_g$ , there exists at least one action transformer policy  $\pi_g \in \Pi_g$ , from our  
505 definition of  $\mathcal{P}_g$ . Let  $\text{RL} \circ \text{ATIRL}_\psi(P_{real})$  lead to a policy  $\tilde{\pi}_g$ , with a marginal transition distribution  
506  $\tilde{\rho}_g$ . The marginal transition distribution induced by  $\overline{\text{RL}} \circ \overline{\text{ATIRL}}_\psi(P_{real})$  is  $\bar{\rho}_g$ .

507 We need to prove that  $\tilde{\rho}_g = \bar{\rho}_g$ , and we do so by contradiction. We assume that  $\tilde{\rho}_g \neq \bar{\rho}_g$ . For this  
508 inequality to be true, the marginal transition distribution of the result of  $\text{RL}(\tilde{c})$  must be different than  
509 the result of  $\overline{\text{RL}}(\bar{c})$ , or the cost functions  $\tilde{c}$  and  $\bar{c}$  must be different.

510 Let us compare the RL procedures first. Assume that  $\tilde{c} = \bar{c}$ .

$$\begin{aligned}\text{RL}(\tilde{c}) &= \operatorname{argmin}_{\pi} \mathbb{E}_{\rho_g} [\tilde{c}(s, a, s')] \\ &= \operatorname{argmin}_{\rho_g} \mathbb{E}_{\rho_g} [\tilde{c}(s, a, s')] \quad \dots (\text{surjective mapping}) \\ &= \overline{\text{RL}}(\bar{c})(\tilde{c} = \bar{c})\end{aligned}$$

511 which leads to a contradiction.

512 Now let's consider the cost functions presented by  $\text{ATIRL}_\psi(P_{real})$  and  $\overline{\text{ATIRL}}_\psi(P_{real})$ . Since  $\text{RL}(\tilde{c})$   
513 and  $\overline{\text{RL}}(\bar{c})$  lead to the same marginal transition distributions, for the inequality we assumed at the  
514 beginning of this proof to be true,  $\text{ATIRL}_\psi(P_{real})$  and  $\overline{\text{ATIRL}}_\psi(P_{real})$  must return different cost  
515 functions.

$$\begin{aligned}
\text{ATIRL}_\psi(P_{real}) &= \operatorname{argmax}_{c \in \mathcal{C}} -\psi(c) + \left( \min_{\pi_g} \mathbb{E}_{P_g} [c(s, a, s')] \right) - \mathbb{E}_{P_{real}} [c(s, a, s')] \\
&= \operatorname{argmax}_{c \in \mathcal{C}} -\psi(c) + \left( \min_{\pi_g} \sum_{s, a, s'} \rho_g(s, a, s') c(s, a, s') \right) - \\
&\quad \sum_{s, a, s'} \rho_{real}(s, a, s') c(s, a, s') \\
&= \operatorname{argmax}_{c \in \mathcal{C}} -\psi(c) + \left( \min_{\rho_g} \sum_{s, a, s'} \rho_g(s, a, s') c(s, a, s') \right) - \\
&\quad \sum_{s, a, s'} \rho_{real}(s, a, s') c(s, a, s') \\
&= \overline{\text{ATIRL}}_\psi(P_{real})
\end{aligned}$$

516 which leads to another contradiction. Therefore, we can say that  $\bar{\rho}_g = \rho_{\tilde{g}}$ .  $\square$

#### 517 **B.4 Proof of Lemma 4.2**

518 We prove convexity under a particular agent policy  $\pi$  but across AT policies  $\pi_g \in \Pi_g$

519 **Lemma B.1.**  $\mathcal{P}_g$  is compact and convex.

520 *Proof.* We first prove convexity of  $\rho_{\Pi_g, t}$  for  $\pi_g \in \Pi_g$  and  $0 \leq t < \infty$ , by means of induction.

521 Base case:  $\lambda \rho_{at_1, 0} + (1 - \lambda) \rho_{at_2, 0} \in \rho_{\Pi_g, 0}$ , for  $0 \leq \lambda \leq 1$ .

$$\begin{aligned}
\lambda \rho_{at_1, 0}(s, a, s') + (1 - \lambda) \rho_{at_2, 0}(s, a, s') &= \lambda \rho_0(s) \pi(a|s) \sum_{\tilde{a} \in \mathcal{A}} \pi_{at_1}(\tilde{a}|s, a) P_{sim}(s'|s, \tilde{a}) \\
&\quad + (1 - \lambda) \rho_0(s) \pi(a|s) \sum_{\tilde{a} \in \mathcal{A}} \pi_{at_2}(\tilde{a}|s, a) P_{sim}(s'|s, \tilde{a}) \\
&= \rho_0(s) \pi(a|s) \sum_{\tilde{a} \in \mathcal{A}} (\lambda \pi_{at_1}(\tilde{a}|s, a) + (1 - \lambda) \pi_{at_2}(\tilde{a}|s, a)) P_{sim}(s'|s, \tilde{a})
\end{aligned}$$

522  $\Pi_g$  is convex and hence  $\rho_0(s) \pi(a|s) \sum_{\tilde{a} \in \mathcal{A}} (\lambda \pi_{at_1}(\tilde{a}|s, a) + (1 - \lambda) \pi_{at_2}(\tilde{a}|s, a)) P_{sim}(s'|s, \tilde{a})$  is  
523 a valid distribution, meaning  $\rho_{\Pi_g, 0}$  is convex.

524 Induction Step: If  $\rho_{\Pi_g, t-1}$  is convex,  $\rho_{\Pi_g, t}$  is convex.

525 If  $\rho_{\Pi_g, t-1}$  is convex,  $\lambda \rho_{at_1, t}(s) + (1 - \lambda) \rho_{at_2, t}(s)$  is a valid distribution. This is true simply by  
526 summing the distribution at time  $t - 1$  over states and actions.

$$\begin{aligned}
\lambda \rho_{at_1, t}(s, a, s') + (1 - \lambda) \rho_{at_2, t}(s, a, s') &= \lambda \rho_{at_1, t}(s) \pi(a|s) \sum_{\tilde{a} \in \mathcal{A}} \pi_{at_1}(\tilde{a}|s, a) P_{sim}(s'|s, \tilde{a}) \\
&\quad + (1 - \lambda) \rho_{at_2, t}(s) \pi(a|s) \sum_{\tilde{a} \in \mathcal{A}} \pi_{at_2}(\tilde{a}|s, a) P_{sim}(s'|s, \tilde{a}) \\
&= (\lambda \rho_{at_1, t}(s) + (1 - \lambda) \rho_{at_2, t}(s)) \pi(a|s) \\
&\quad \sum_{\tilde{a} \in \mathcal{A}} (\lambda \pi_{at_1}(\tilde{a}|s, a) + (1 - \lambda) \pi_{at_2}(\tilde{a}|s, a)) P_{sim}(s'|s, \tilde{a})
\end{aligned}$$

527  $\lambda \rho_{at_1, t}^\pi(s) + (1 - \lambda) \rho_{at_2, t}^\pi(s)$  is a valid distribution, and  $\Pi_g$  is convex. This proves that the transition  
528 distribution at each time step is convex. The normalized discounted sum of convex sets (Equation 9)  
529 is also convex. Since the exponential discounting factor  $\gamma \in [0, 1)$ , the sum is bounded as well.  $\square$

530 We now prove Lemma 4.2.

531 **Lemma 4.2.**  $\overline{\text{RL}} \circ \overline{\text{ATIRL}}_\psi(P_{\text{real}}) = \operatorname{argmin}_{\rho_g \in \mathcal{P}_g} \psi^*(\rho_g - \rho_{\text{real}})$ .

532 *Proof of Lemma 4.2* Let  $\bar{c} = \overline{\text{ATIRL}}(P_{\text{real}})$ ,  $\bar{\rho}_g = \overline{\text{RL}}(\bar{c}) = \overline{\text{RL}} \circ \overline{\text{ATIRL}}(P_{\text{real}})$  and

$$\hat{\rho}_g = \operatorname{argmin}_{\rho_g} \psi^*(\rho_g - \rho_{\text{real}}) = \operatorname{argmin}_{\rho_g} \max_c -\psi(c) + \sum_{s,a,s'} (\rho_g(s,a,s') - \rho_{\text{real}}(s,a,s'))c(s,a,s') \quad (14)$$

533 where  $\psi^* : \mathcal{C}^* \mapsto \mathbb{R}$  is the convex conjugate of  $\psi$ , defined as  $\psi^*(c^*) := \sup_{c \in \mathcal{C}} \langle c^*, c \rangle - \psi(c)$ .  
 534 Applying the above definition to the rightmost term in the above equation gives us the middle term.

535 We now argue that  $\bar{\rho}_g = \hat{\rho}_g$  which are the two sides of the equation we want to prove. Let us consider  
 536 loss function  $L : \mathcal{P}_g \times \mathbb{R}^{\mathcal{S} \times \mathcal{A} \times \mathcal{S}} \mapsto \mathbb{R}$  to be

$$L(\rho_g, c) = -\psi(c) + \sum_{s,a,s'} (\rho_g(s,a,s') - \rho_{\text{real}}(s,a,s'))c(s,a,s') \quad (15)$$

537 We can then pose the above formulations as:

$$\hat{\rho}_g \in \operatorname{argmin}_{\rho_g \in \mathcal{P}_g} \max_c L(\rho_g, c) \quad (16)$$

$$\bar{c} \in \operatorname{argmax}_c \min_{\rho_g \in \mathcal{P}_g} L(\rho_g, c) \quad (17)$$

$$\bar{\rho}_g \in \operatorname{argmin}_{\rho_g \in \mathcal{P}_g} L(\rho_g, \bar{c}) \quad (18)$$

538  $\mathcal{P}_g$  is compact and convex (by Lemma B.1) and  $\mathbb{R}^{\mathcal{S} \times \mathcal{A} \times \mathcal{S}}$  is convex.  $L(\cdot, c)$  is convex over all  $c$  and  
 539  $L(\rho_g, \cdot)$  is concave over all  $\rho_g$ . Therefore, based on minimax duality:

$$\min_{\rho_g \in \mathcal{P}_g} \max_c L(\rho_g, c) = \max_c \min_{\rho_g \in \mathcal{P}_g} L(\rho_g, c) \quad (19)$$

540 From Equations 16 and 17,  $(\hat{\rho}_g, \bar{c})$  is a saddle point of  $L$ , implying  $\hat{\rho}_g = \operatorname{argmin}_{\rho_g \in \mathcal{P}_g} L(\rho_g, \bar{c})$  and  
 541 so  $\bar{\rho}_g = \hat{\rho}_g$ .

542 □

## 543 B.5 Proof of Lemma 4.3

544 **Lemma 4.3.** *The marginal transition distribution of  $\operatorname{argmin}_{\pi_g} \psi^*(\rho_g - \rho_{\text{real}})$  is equal to*  
 545  $\operatorname{argmin}_{\rho_g \in \mathcal{P}_g} \psi^*(\rho_g - \rho_{\text{real}})$ .

546 *Proof.* The proof of equivalence here is simply to prove that optimizing over  $\pi_g$  is the same as  
 547 optimizing over  $\rho_g$ . From Equation 10 and from the fact that agent policy  $\pi$  and simulator transition  
 548 function  $P_{\text{sim}}$  are fixed, we can say that the only way to optimize  $\rho_g$  is to optimize  $\pi_g$ , which leads  
 549 to the above equivalence. □

## 550 C Experimental Details

551 To collect expert trajectories from the real world, we rollout the stochastic initial policy trained  
 552 in sim for 1 million timesteps, on the real world. This dataset serves as the expert dataset during  
 553 the imitation learning step of GARAT. At each GAN iteration, we sample a batch of data from the  
 554 grounded simulator and expert dataset and update the discriminator. Similarly, we rollout the action  
 555 transformer policy in its environment and update  $\pi_g$ . We perform 50 such GAN updates to ground

Name	Value
Hidden Layers	2
Hidden layer size	64
timesteps per batch	5000
max KL constraint	0.01
$\lambda$	0.97
$\gamma$	0.995
learning rate	0.0004
cg damping	0.1
cg iters	20
value function step size	0.001
value function iters	5

Table 1: Hyperparameters for the TRPO algorithm used to update the Agent Policy

Name	Value
Hidden Layers	2
Hidden layer size	64
nminibatches	2
Num epochs	1
$\lambda$	0.95
$\gamma$	0.99
clipping ratio	0.1
time steps	5000
learning rate	0.0003

Table 2: Hyperparameters for the PPO algorithm used to update the Action Transformer Policy

the simulator using GARAT. The hyperparameters for the PPO algorithm used to update the action transformer policy is provided in Table 2. The hyperparameters used for the TRPO algorithm to update the agent policy can be found in Table 1.

We implemented different IFO algorithms and noticed that there was no significant difference between these backend algorithms in sim-to-real performance. During the discriminator update step in GAIfO-reverseKL (AIRL), GAIfO and GAIfO-W (WAIL), we use two regularizers in its loss function - L2 regularization of the discriminator’s weights and a gradient penalty (GP) term, with a coefficient of 10. Adding the GP term has been shown to be helpful in stabilizing GAN training [22].

In our implementation of the AIRL [10] algorithm, we do not use the special form of the discriminator, described in the paper, because our goal is to simply imitate the expert and does not require recovering the reward function as was the objective of that work. We instead use the approach Ghasemipour et al. [12] use with state-only version of AIRL.

GAT uses a smoothing parameter  $\alpha$ , which we set to 0.95 as suggested by Hanna and Stone [14]. RARL has a hyperparameter on the maximum action ratio allowed to the adversary, which measures how much the adversary can disrupt the agent’s actions. This hyperparameter is chosen by a coarse grid-search. For each domain, we choose the best result and report the average return over five policies trained with those hyperparameters. We used the official implementation of RARL provided by the authors for the MuJoCo environments. However, since their official code does not readily support PyBullet environments, for the Ant and Minitaur domain, we use our own implementation of RARL, which we reimplemented to the best of our ability. When training a robust policy using Action space Noise Envelope (ANE), we do not know the right amount of noise to inject into the agent’s actions. Hence, in our analysis, we perform a sweep across zero mean gaussian noise with multiple standard deviation values and report the highest return achieved in the target domain with the best hyperparameter, averaged across 5 different random seeds.

Environment Name	Property Modified	Default Value	Modified Value
InvertedPendulumHeavy	Pendulum mass	4.89	100.0
HopperHeavy	Torso Mass	3.53	6.0
HopperHighFriction	Foot Friction	2.0	2.2
HalfCheetahHeavy	Total Mass	14	20
WalkerHeavy	Torso Mass	3.534	10.0
Ant	Gravity	-4.91	-9.81
Minitaur [38]	Torque vs. Current	linear	non-linear

Table 3: Details of the Modified Sim-to-“Real” environments for benchmarking GARAT against other black-box Sim-to-Real algorithms.

### 580 C.1 Modified environments

581 We evaluate GARAT against several algorithms in the domains shown in Figure 3. Table 3 shows the  
582 source domain along with the specific properties of the environment/agent modified. We modified  
583 the values such that a policy trained in the sim environment is unable to achieve similar returns in  
584 the modified environment. By modifying an environment, we incur the risk that the environment  
585 may become too hard for the agent to solve. We ensure this is not the case by training a policy  $\pi_{real}$   
586 directly in the “real” environment and verifying that it solves the task.

### 587 C.2 Simulator Grounding Experimental Details

588 In Section 6.1, we show results which validate our hypothesis that GARAT learns an action trans-  
589 formation policy which grounds the simulator better than GAT. Here we detail our experiments for  
590 Figure 1.

591 In Figure 1a, we plot the average error in transitions in simulators grounded with GARAT and GAT  
592 with different amounts of “real” data, collected by deploying  $\pi$  in the “real” environment. The  
593 per step transition error is calculated by resetting the simulator state to states seen in the “real”  
594 environment, taking the same action, and then measuring the error in the L2-norm with respect  
595 to “real” environment transitions. Figure 1a shows that with a single trajectory from the “real”  
596 environment, GARAT learns an action transformation that has similar average error in transitions  
597 compared to GAT with 100 trajectories of “real” environment data to learn from.

598 In Figure 1b, we compare GARAT and GAT more qualitatively. We deploy the agent policy  $\pi$  from  
599 the same start state in the “real” environment, the simulator, GAT-grounded simulator, and GARAT-  
600 grounded simulator. Their resultant trajectories in one of the domain features (angular position of the  
601 pendulum) is plotted in Figure 1b. The trajectories in GARAT-grounded simulator keeps close to the

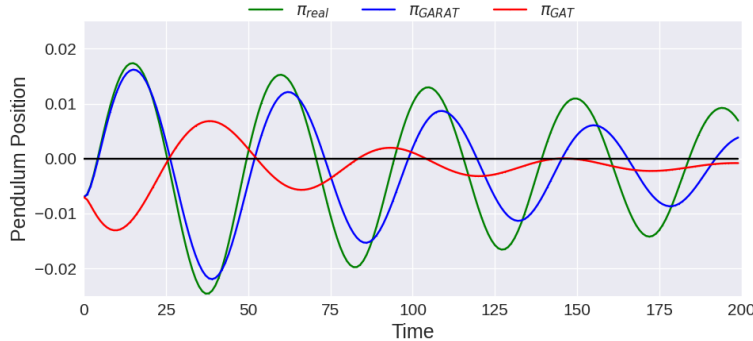


Figure 4: Policies trained in “real” environment, GAT-grounded simulator, and GARAT-grounded simulator deployed in the “real” environment from the same starting state

602 “real” environment, which neither the ungrounded simulator nor the GAT-grounded simulator manage.  
603 The trajectory in the GAT-grounded simulator can be seen close to the one in the “real” environment  
604 initially, but since it disregards the sequential nature of the problem, the compounding errors cause  
605 the episode to terminate prematurely.

606 An additional experiment we conducted was to compare the policies trained in the “real” environment,  
607 GAT-grounded simulator and GARAT-grounded simulator. This comparison is done by deploying  
608 them in the “real” environment from the same initial state. As we can see in Figure 4, the policies  
609 trained in the “real” environment and the GARAT-grounded simulator behave similarly, while the one  
610 trained in the GAT-grounded simulator acts differently. This comparison is another qualitative one.  
611 How well these policies perform in w.r.t. the task at hand is explored in detail in Section 6.2