

# Tug-Of-War Model for Two-Bandit Problem (POSTER)

Song-Ju Kim<sup>1</sup>, Masashi Aono<sup>2</sup>, and Masahiko Hara<sup>1,2</sup>

<sup>1</sup> Fusion Technology Center 5F, Hanyang University,  
17 Haengdang-dong, Seongdong-gu, Seoul 133-791, Korea

<sup>2</sup> 2-1 Hirosawa, Wako-shi, Saitama 351-0198, Japan  
{songju,masashi.aono,masahara}@riken.jp

Flucto-Order Functions Asian Collaboration Team,  
RIKEN Advanced Science Institute

**Abstract.** The amoeba of the true slime mold *Physarum polycephalum* shows high computational capabilities. In the so-called amoeba-based computing, some computing tasks including combinatorial optimization are performed by the amoeba instead of a digital computer. We expect that there must be problems living organisms are good at solving. The “multi-armed bandit problem” would be the one of such problems. Consider a number of slot machines. Each of the machines has an arm which gives a player a reward with a certain probability when pulled. The problem is to determine the optimal strategy for maximizing the total reward sum after a certain number of trials. To maximize the total reward sum, it is necessary to judge correctly and quickly which machine has the highest reward probability. Therefore, the player should explore many machines to gather much knowledge on which machine is the best, but should not fail to exploit the reward from the known best machine. We consider that living organisms follow some efficient method to solve the problem.

We propose a model, named as the “tug-of-war (TOW) model,” based on the photoavoidance behavior of amoeba induced by light stimuli. The TOW model is a bio-inspired computing method capable of solving the problem efficiently, while it is not necessarily a biological model for reproducing a behavior of the true slime mold. In this study, we focus on the two-bandit problem. In the original version of the problem, the player can select only one machine for a trial. To explore advantages of parallel computing, we extend this problem so that the player is allowed to select both machines for a trial. We show that the TOW model exhibits good performance. The average accuracy rate of the TOW model is higher than that of well-known algorithms such as the modified  $\epsilon$ -greedy algorithm and the modified softmax algorithm. Additionally, the TOW model is effective for solving relatively difficult problems in which the reward probabilities of the two machines are close. Finally, we show that the TOW model is good at adapting to a changing environment in which the reward probabilities are dynamically updated. The results are highly suggestive because, in order to survive in the unknown world, living organisms have to adapt to new environments as quickly as possible at the expense of a slight reduction in accuracy.