

Notizen zum Skype Meeting vom 19.11.2015

von Basti und Sarah

9. Dezember 2015

Kapitel 1

Distributed Recognition

1.1 Allgemein/Ablauf

- Flos Arbeit über nestmate recognition als Vorarbeit
- gemeinsame VÖ über TruDiPhe Anfang 2016 (Frage: Ideen für Konferenz?)
- dann DFG-Antrag für 2 Arbeiter

1.2 Inhalte

2 Hauptansatzpunkte möglich:

1. Nestmate recognition (NR)

- erkennen als freundlich → positiv
- bekämpfen/feindlich → negativ

Aber: nicht feiner differenzierbar, da Ameisen sich in der Situation, wenn sie anderer Ameise gegenüberstehen, sofort entscheiden müssen, ob sie angreifen oder nicht.

2. Footprints (FP)

für Flo interessanter, da facettenreicher (Spatial Patterns through footprint response?). Ameisen können Fußspuren

- folgen → positiv
- ignorieren → neutral
- meiden → negativ

Mehr Möglichkeiten, Verhalten abzuwägen zB auch starker/schwacher Drang, den Fußspuren zu folgen.

Ausgangslage Basti/Sarah

3 Teile:

- **Trusted Desktop Grid:** Simulation von Multi-Agenten-System, abstrakte Jobs, die in Work Units gesplittet werden und verteilt gerechnet werden. Ziel ist, speedup zu maximieren (von einzelner Agenten und Gesamtsystem) $speedup = \frac{time_{self}}{time_{distributed}}$, also: wie lang brauch ich wenn ich alles selber rechne im Vergleich zu wenn ichs verteile?. Natürlich müssen die Agenten dann aber auch für andere arbeiten, um sich Vertrauen zu verdienen, denn sonst nimmt Dir auch keiner Deine Arbeit ab. Es gibt Trust Communities (TCs), Zusammenschluss von sich gegenseitig stark vertrauenden Agenten
- **Trusted Ants:** Simulation in Unity, TDG in "schön"/ nachvollziehbar/ leichter zu verstehen; hier sind Jobs Wegpunkte, die abgelaufen werden müssen
- **T-Grid:** Real-Implementierung des TDG, hoffentlich im Februar in erster, lauffähiger Prototyp. Ist Software, die auf mehrere Rechner gespielt wird, diese können sich dann, wenn sie vertrauen, gegenseitig Rechenleistung abtreten, wenn sie grad nicht genutzt wird.

Problem bei allen drei: zentrale Datenbank für Reputationswerte (Reputation $\hat{=}$ aggregierter Trust). Z.B. in TC bei Rust Community Manager (TCM) Als erster Lösungsansatz gibt es das Konzept der **Digitalen Trust Pheromone** (TruDiPhe), d.h., Trust-Werte werden bei dem jeweiligen Adressat dieses Wertes verschlüsselt gespeichert.

Fragen/ Zum Nachdenken

Eher Basti/Sarah:

Wie mappen wir das alles auf NR/FP?

Mappt Job**annahme** auf FP? Sollen wir die Effektivität klassifizieren? Also bei T-Grid, so dass das Grid gut ausgelastet und zielführend (wichtige Sachen zuerst?) benutzt wird? Erweiterung der Überlegungen auf Arbeitgeben (submitter)! Prios einführen bei Arbeiten!

Also: Mappt „Kriterium zielführend“ auf „Art der FP“? Mapping Pfad \leftrightarrow Aufgabe

NR: wie Schablone, die man anpasst \rightarrow zu trivial

Radius eingrenzen \leftrightarrow Größe der TC

Lernen! anfangs misstrauisch am besten? Vertrauensvorschuss gewähren?

Eher Flo:

Zusammenhang FP und Umwelteinflüsse?

Neugier modellieren?

Wie schnell verschwindet hydrocarbon? Ameisen nehmen Stärke FP in Entscheidung auf, aber können nicht unterscheiden ob neu und schwach oder alt und intensiv gewesen.

Überlagerung von FP

FP ignorieren/alleine Futter finden \leftrightarrow submitter können auch selber rechnen (B&S)

bei Vertrauensbruch Fußspuren weniger folgen / weniger explorativ

Beide:

Arbeitsvergabe (B&S) \leftrightarrow Suche der Fußspuren (F) !!! (X)

Systemperformanz als Metrik?

Evolution - Nachkommen erben Parameter

In offenem System selbst-organisierte Effektivität; ohne zentrale Kontrolle beste Pfade zu beschreiben rausfinden

Optimale Strategie finden, die wir dann anwenden. In TDG: pro Agentenzusammensetzung optimale Strategie zur Laufzeit lernen. quantitativ: Parametergrenzen identifizieren und allgemein was daraus ableiten;

1.3 ToDo

- working title finden
- Gemeinsame Zielsetzung formulieren
- Zeitplan erstellen
- ...?

Plans Augsburg

For this project we want to combine the biological concepts of hydrocarbon footprints (FP) (and nestmate recognition (NR)) (Mainz) with the Organic Computing domain, especially the distributed verification of trust (values) in open, heterogeneous multi-agent systems (MAS) (Augsburg) to have a mutual benefit.

(explain concept NR, FP)

(explain concept trust/MAS - Trusted Ants (TA))

In general, we can map nestmates (well-behaving, WB) and ants from foreign colonies (which may be avoided or ignored; bad-behaving, BB) to WB, BB (,or something in between) agents/PCs/etc.

TA can profit from FP in several ways. A first mapping that can be useful for us is to map WB/BB ants to legal/illegal jobs in a MAS. One question that we ask—but cannot answer yet—is the territorial question. Regarding, e.g. legal and illegal jobs that may be processed distributedly. If an entity worked for a BB entity, it maybe wants to avoid similar entities from that moment on. In the ant domain this means to avoid the territory, and thus the footprints, of ants; in the OC domain it means to avoid jobs from this or similar users. Here, we also have to ask the question: what is „similar“? How can our similarity be mapped to NR similarity in terms of territoriality? To answer that is part of the future work.

Another aspects from FP can be mapped to the TA domain: an ant colony can be mapped to a job in TA, in consequence, a (group of) ant maps to a work unit. Different ant colonies represent jobs with different prioritisations. The aim (from OC side) here would be to maximise the utilisation¹ of the system. How do ants change their tracking behaviour² after conflict confrontation, or positive experiences? This conflict can be mapped to a work unit that was not processed correctly by a fellow ant. How much should the behaviour of the affected ant (submitter) be touched by that, i.e. how much should its trust/distribution of work units change after that? The outcome of FP experiments are interesting for us, since the tracking behaviour seems successful on ants, and offer new perspectives about what options, e.g. what strength of Δ_{trust} is purposeful?

Another approach is, if we map an ant colony to a Trust Community (TC). Inside the group, entities are sure everything is fine. With the preliminary work of Menzel et al. we can apply mechanisms of NR, e.g. co-existence of species³, to systems for a distributed mechanism of building TCs. ⁴

How can the territorial aspect in FP(/NR) be meaningfully used in TA? How does it improve our TA/TDG/simulations on distributed verification/trust?⁵

Why are our simulations useful for FP research?

¹(german: Auslastung... wording?)

²Flo: wording correct?

³Flo: wie ist da der Fachterminus?

⁴Flo: ist there also some kind of splitting possible in a species? Or does this only happen if a queen founds a new colony?

⁵Not sure about that... is that what we already have to write/know in the proposal? Or part of the work we want to do?

1.4 Mapping

Mapping 1: FPs als Jobs. Die Ameisen sind gleichzusetzen mit Agenten, die Arbeiten ausführen sollen. Die Ausführung einer Aufgabe ist äquivalent zum Beschreiten eines Pfads, ihre Erledigung mit dem Ankommen. Konkret könnte man das Verfolgen eines Pfads als Annehmen und Bearbeiten eines Jobs interpretieren. FPs markieren einen Pfad. Entsprechend können sie Agenten motivieren ihm zu folgen. Je nach Aufgabe, je nach Agent, können die FPs dazu führen, dass Ameisen kein Interesse an dieser Route haben, darin bestärkt werden, ihr zu folgen, oder sie aktiv vermeiden. Letztendlich muss ein FP eine Ameise davon überzeugen, dass es die richtige Entscheidung ist, hier Zeit und Energie zu investieren. Trust-Deposits von vorausgehenden Ameisen, die beschreiben, wer sie sind, warum sie das tun, wie toll sie es finden, oder Ähnliches können dazu führen, dass man mitmacht oder, falls man andere Erfahrungen gemacht oder andere Ziele hat, dass man den Pfad meidet oder ignoriert. Meiden hieße, dass man proaktiv nach Alternativen sucht. Ignorieren hieße, dass man durchaus ein paar Schritte den Pfad entlang laufen könnte, aber nicht wirklich motiviert, eher zufällig. Das mehrfache Folgen eines Pfads wäre gleichbedeutend mit einem mehrfachen oder verteilten Abarbeiten eines Jobs. Zudem könnte es sein, dass eine Ameise den Pfad frühzeitig verlässt, aufgrund von besseren Möglichkeiten, die erst im Lauf der Zeit auftreten (Kreuzung von FPs), oder aufgrund von Konflikten, die in der Arbeit erkannt werden (Überlagerung mit negativen FPs). In technischen Systemen wären die Analogien bspw. einfachere Jobs mit höherem Benefit, Feststellen der Unerfüllbarkeit eines Jobs, oder Feststellen der Illegalität eines Jobs. Eine andere Ameise, die andere Fähigkeiten hat oder weniger Gewissensbisse, könnte einen verlassenen Pfad weiterlaufen. Soweit zum Folgen/Vermeiden: wie aber Entstehen die ersten FPs? Der User könnte eine Aufgabe in den Raum stellen. Eine oder mehrere Ameisen nehmen sie an (je nach Job, je nach Systemkonfiguration), arbeiten sie ab und hinterlassen dabei einen Pfad. Der Pfad mag sinnvoll oder weniger sinnvoll sein, das müssen nachfolgende Ameisen über die FPs entscheiden. Alternativ kann man sich auch vorstellen, dass der User initiale FPs vorgibt.

Mapping 2: FPs als Ressourcen Bewertungen. Aus Sicht eines Agenten, der einen oder mehrere Jobs erledigen möchte, hat man eine "Landschaft" von Computing Resources (Infrastruktur) zur Verfügung und möchte nun wissen, wo man seine Jobs ausführen lassen soll. Vorhergehende Agenten, die die Infrastruktur mit verschiedenen Ansprüchen genutzt haben, können Bewertungen hinterlegt haben bzw. die Resources mögen sie akkumuliert haben und anfragende Agenten damit umwerben. Je nachdem, was die Agenten sehen, folgen sie den Angeboten, ignorieren sie (Umsetzung bspw. als Nutzung mit 50% Wahrscheinlichkeit) oder suchen proaktiv nach Alternativen.