UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2004

MEng Honours Degree in Information Systems Engineering Part IV
MSci Honours Degree in Mathematics and Computer Science Part IV
MEng Honours Degrees in Computing Part IV
MSc in Advanced Computing
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the Associateship of the City and Guilds of London Institute
This paper is also taken for the relevant examinations for the Associateship of the Royal College of Science

PAPER C474=I4.8

MULTI-AGENT SYSTEMS

Monday 26 April 2004, 14:30
Duration: 120 minutes

Answer THREE questions

Paper contains 4 questions
Calculators not required

1a Briefly describe the key features of the *contract net* protocol for allocating tasks to a network of agents/problem solvers. Include descriptions of:

i) the activities of a node as a manager

ii) the activities of a node as a potential contractor

iii) the content of a contract announcement

iv) the content of a bid

v) immediate response announcements, why they might be used, and their possible replies

In your answer to iii) explain the use of the eligibility and bid specification fields, and in your answer to iv) explain the role of the node abstraction field.

b A mobile agent is an agent that can move from one host to another. On each host it might interact with local agents – either static agents or other mobile agents.

i) What support software is needed to allow a mobile agent to move from host to host and to discover local agents with which it might want to interact?

ii) How might a mobile agent be moved?

c Suggest ways in which mobile agents, fixed personal assistance agents, and the contract net protocol might be used to implement a workflow system.

The three parts carry, respectively, 50%, 25%, 25% of the marks.

2. Compare the agent architecture of the Agent0 agent programming language with that of AgentSpeak(L). In particular:

i) Discuss the similarities and differences in internal state.

ii) Explain how each type of agent is programmed.

iii) Explain how new information and requests for action are communicated between agents in each language.

iv) Briefly compare the execution cycles of each type of agent.

v) If an AgentSpeak(L) plan had associated meta-information about what primitive actions it contained and/or what resources it would use, what feature of Agent0 might be usefully incorporated into the AgentSpeak(L) language to give a means of programming the option selection function? Briefly justify your answer.

3a Briefly explain the role of a *matchmaker* in a distributed information system.

b Explain, with one example for each, the use of the KQML performatives:

ask-all, stream-all, subscribe.

State what form of message or messages the receiving agent will use to respond to your example messages. How will the response message be linked to the initial message?

c Below is a QuProlog program which defines a simple matchmaker. The program assumes that a KQML message of the form:

(performative :attr1 v1 :attr2 v2 .... :attrk vk)

is represented as the Prolog term:

performative([attr1(v1), attr2(v2),..., attrk(vk)])

where any '-' in the KQML performative or attribute name is replaced by '_', all attribute values are Prolog terms. and the sender and receiver attributes are dropped. The program can handle *advertise* and *recommend* messages using an unspecified match/2 relation to match their content messages.

```
?-dynamic(advertised/2).
match_maker :-
        message_choice (
        advertise(AttrsList) <<- S ->
                member(content(KQMLMess),AttrsList),
                assert(advertised(S,KQMLMess))
        ;
        recommend(AttrsList) <<- S  ->
                member(content(KQMLMess1,AttrsList),
                member(reply_with(L),AttrsList),
                advertised(Ag,KQMLMess2),
                match(KQMLMess1,KQMLMess2),
                reply([in_reply_to(L),content(Ag)]) ->> S),
        match_maker.
```

i) Assume that the match/2 relation will successfully match content messages that use the same performative and otherwise just mention an ontology and a content syntax. Give the Prolog term message that would be sent to *advertise* that the sender can respond to *stream_all* queries using the *ontology* folklore and 'Prolog' as the query *syntax.*

ii) Give the changes you need to make to the above program so that it can also accept KQML *subscribe* messages containing as content a *recommend* message of the form it can already handle.

The three parts carry, respectively, 10%, 35%, 55% of the marks.

4a

i) Define the concept of Pareto optimality.

(i) Define the concept of Nash equilibrium.

iii) When designing negotiation mechanisms for multi-agent systems, we would typically like these mechanisms to be both stable (i.e. have a Nash equilibrium) and efficient (i.e. guarantee Pareto optimal outcomes). However, in practice, this is often not possible. Give an example of such a conflict and briefly explain your answer. (A detailed description of the game/mechanism is not required.)

b Consider a scenario where two agents, agent 1 and agent 2, negotiate over the exchange of a number of resources using the Monotonic Concession Protocol. Suppose, in round n, agent 1 makes the proposal $x1$ and agent 2 makes the proposal $x2$.

Recall the formula used to evaluate agent 1's willingness to risk conflict, as used for the Zeuthen Strategy ($u1$ is the agent's utility function):

$$risk1(x1, x2) = \frac{u1(x1) - u1(x2)}{u1(x1)}$$

i) Briefly explain the intuition behind this formula.

ii) Briefly explain how the formula is used in the Zeuthen Strategy to decide whose turn it is to concede next.

c i) Define the protocol of the Vickrey auction mechanism.

ii) What is the dominant strategy for a bidder in a private value Vickrey auction?

iii) Briefly justify your answer to part (ii).

The three parts carry, respectively 40%, 20%, and 40% of the marks.