Agreement–Cooperation Study: Technical Setup

Version 1.0 Provisional, June 30, 2007
Passage to *Version 1.0 Definite* planned for October 31, 2007, after the Patras COST meeting.

Eric Keller, IMM, Lettres, University of Lausanne

Contents

Abstract

This text introduces and motivates the technical setup for the Agreement-Cooperation study, to be performed in the framework of Working Group 1 of the COST 2102 European project. It contains a summary review of the project, the experiment and the recording equipment; it reviews the motivations leading to the principal specifications of the experiment, summarizes the experimental paradigm, specifies various recording settings, describes the synchronized recording programme, and reviews the requirements for the conversation game to be played in the experiment.

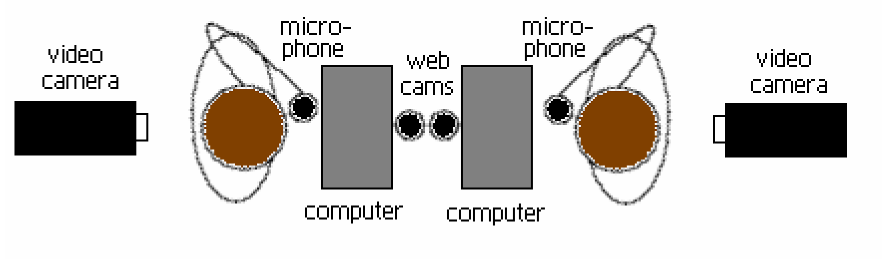
1. *Introduction*

1.0. Objectives

The Agreement-Cooperation Study intends to examine prosodic, phonetic, facial and gestural components associated with the occurrence of agreement and disagreement in conversations, performed either in a cooperative or non-cooperative environment. The project focuses a multi-approach, multi-centre and multi-cultural analysis onto the well-circumscribed and theoretically important issue of the evolution of cooperation and social support within conversations. A key aspect of the study is the examination of visual and speech parameters in the evolution of cooperative and of uncooperative behaviour between game players when different payoff situations exist. An important long-term objective of the project is the possibility of creating an automatic system for evaluating agreement, cooperation and support within conversations, to be used in enterprise training or in family therapy. A review and a basic motivation for the study are given in Keller & Tschacher (in press).

The experiment will attempt to provoke various agreement/disagreement conditions within both cooperative and uncooperative situations. It is known that concrete instances of verbal agreement or disagreement have different social effects if they are manifested within a tacit cooperative or supporting climate than if they are expressed in an uncooperative or conflictual climate. For this reason, two subjects will be induced by a manipulation of the rewards of the conversational game to be cooperative in one part of the experiment, and they will be induced to be uncooperative in another part of the experiment. We hope that the experimental manipulation in this study will set up various combinations of tacit cooperative vs. non-cooperative environments, which in turn should give rise to concrete manifestations of agreement and non-agreement behaviour.

The purpose of the present document is to describe the technical setup that will be necessary to record the speakers. The two subjects are seated directly opposite each other in front of two computers (or two computer keyboards and two computer screens) (Figure 1)[1]. Each subject is wearing a microphone around the neck (not on the ears); a high-resolution web camera sitting on top of the screen (about a meter in front of the subject) records facial gestures. Two digital cameras situated high behind the subjects make waist-up recordings of the facing subjects.



[1] We originally considered seating the two subjects at a 90-degree angle. However, a pilot recording showed that this caused the subjects to turn their heads away from the cameras. The face-to-face arrangement permits full-face recordings.

Figure 1. Proposed experimental setup: Two subjects are seated opposite each other in front of two computers and are recorded by a total of four cameras and two noise-cancelling microphones.

1.1. Schedule

The study will run for the duration of the COST 2102 project and will terminate at the end of the project (Nov. 30, 2010). The following major phases are foreseen:

1. June 2007 – October 2007. Evaluation and discussion of experimental conditions and pilot projects, to be discussed at the Patras COST meeting (autumn 2007).
2. November 2007 – October 2008. Record the first set of experiments. Initial results to be discussed at the autumn 2008 COST meeting.
   Firm deadline for the first data set: **August 31, 2008**.
3. November 2008 – October 2009. Record a second set of experiments, if required. Evaluation of intermediate results to be discussed at the autumn 2009 COST meeting.
   Firm deadline for the second data set, if any: **August 31, 2009**.
4. November 2009 – October 2010. Evaluation of final results to be discussed at the final COST meeting, autumn of 2010.

1.2. Laboratories and Languages

Any laboratory participating in COST 2102 can join in any part of the experiment either as a recording team and/or as an evaluation team.

- **Recordings** can be performed in any language. Laboratories that provide recordings are expected to share their data within the COST 2102 group and provide two sets of mark-up:
  - Identification of concrete speech segments considered to express agreement/disagreement/other.
  - Identification of tacit speech portions considered to be cooperative/uncooperative/other.
- **Evaluations** are performed by laboratories within their area of specific competence. As an example, Lausanne will be concentrating on prosodic parameters, while Bern and Mataró have indicated an interest in analyzing facial expressions.

1.3. Data Sharing, Software Sharing

- Recording laboratories are expected to share their marked-up recordings within the COST 2102 group.
- Evaluating laboratories may, but do not have to share their analyzing software. The widespread sharing of software is encouraged. Bern and Lausanne have agreed to make their software available to other COST 2102 laboratories for the duration of the project, and to render it public at the end of the project (GNU license).

2. Recording Equipment

2.0. The Equipment

The initial audio-video setup for recording two interacting speakers with two (relatively powerful) computers was fixed on the basis of previous considerations (Keller, 20.4.2007 and 26.4.2007), and in view of consultations held with a number of study participants (W. Tschacher & F. Ramseyer [Berne], E. Keller & I. Pante [Lausanne], A. Esposito [Salerno], M. Faundez-Zanuy [Mataró], P. Murphy [Limerick], A. Nijholt [Twente] and D.K.J. Heylen [Twente])[2].

The use of two computers running in parallel was motivated technically: only two computers running in parallel can store the high-quality audio-video information we wish to obtain for this study (viz., data acquisition speed and disk access times). Also, hooking up more than one webcam and more than one high-quality microphone to a computer is generally difficult or simply impossible.

The recording setup consists of:

1. ***Microphones.*** Two Plantronics DSP 400 noise-cancelling microphones, USB connection (2 x 45€ in Geneva), for high-quality speech recording. These microphones record natively at 48 kHz. The integrated circuit provides noise cancelling for distant speech sources and it probably also provides integrated anti-aliasing and down-sampling (i.e., these latter functions are unlikely to be in the on-board driver). This makes it possible to record very rapidly and with noise cancelling at either the native 48 kHz, or at a number of correctly anti-aliased other sampling rates. We recommend 22 kHz. Our pilot projects have shown that good-quality recordings can be made when keeping the microphones around the neck, rather than on the ears, provided the sound volume is adjusted correctly.
2. ***Web cams.*** Two Logitech QuickCam STX (640x480 pixels), USB connection (2 x 55€ in Geneva), for recording facial expressions. Native format: I420, which is a progressive (non-interleaved) format. This camera integrates automatic illumination adjustment, which is of importance when recording with existing laboratory lighting.
3. ***Video cameras.*** Two digital video cameras recording to mini-cassette or disk, for recording waist-up gestural expression. For cost reasons, no specific camera is suggested; however it is important that the two cameras record essentially to the same level of quality. Video cameras recording at PAL 720x576 pixels interleaved are in common use. Typical prices for new cameras in Geneva are 250+ € for cameras recording to mini-cassette and 500+ € for cameras recording to hard disk. After the experiment, the recording must be transferred to computer.

---

[2] Everyone who participated in these discussions is heartily thanked for their contributions. Thanks also go to Isaac Pante (graduate assistant, University of Lausanne) and Julien Pourtet (BA student, University of Lausanne), for participating in two pilot recordings and discussing their outcome, and special gratitude goes to Julien Pourtet who in record time (in three weeks) wrote an initial version of an interactive recording programme for this study.

4. ***Computers.*** Relatively fast computers (2 GHz+) with sufficient storage capacity. Can be either portable or stationary.
5. ***Recording software.*** The central recording programme is VirtualDub (available only on the Windows platform) which provides a number of important advantages over other programmes (see below). The most important is the ability to record in native format ***without dropping frames*** on current 2+ GHz computers. Tests have shown (see below) that the recordings can be either performed at 15 fps uncompressed format, or in a compressed format which offers compression rates of 50-90:1 [3]. Other major advantages of the VirtualDub programme are its extensive technical documentation and the possibility of launching the recording from an external control programme.
6. ***Experiment software.*** Software is provided by the Lausanne laboratory to perform the recording of video and audio on the Windows platform, integrated with the presentation and processing of the experiment (a programme called "***Sink***" [4]). The Lausanne programme launches the two recordings on the two computers in parallel and it performs automatic data accumulation during the experiment.

2.1. Objectives

This recording setup was chosen in order to satisfy three central parameters:

1. ***Satisfactory quality for the experimental design.*** Participants in this study wish to perform detailed *visual analyses* with both time-series and detailed facial examination. This means that the facial surface has to be rendered with satisfactory detail, and that the recording of frame timing is very regular (no frame stuttering, a low degree of frame jitter, constant bit rate for those who require it). We also wish to perform detailed *audio analyses* to determine the presence of attitudinal changes in the prosodic traces and in the glottal waveform and to examine measures of inter-subject prosodic agreement. A clear and noise-free audio trace for each speaker is essential, despite the subjects' expected active interactional behaviour. Mutual noise cancelling is of great importance, because we wish to study each contribution to the conversation on a separate channel, despite voice overlaps between speakers[5].
2. ***Inexpensive acquisitions.*** Since COST members from very diversely financed laboratories have expressed their interest in participating, and since there is great urgency to obtain data as soon as possible (COST 2102 ends in November 2010),

[3] Whether the bit rate is constant or not is in all probability not very important for our purposes. What is important is a constant *frame rate* that permits a very close reconstitution of regularly spaced frames from the compressed stream.

[4] A weak take-off on "sync", for "synchronous recording", or an acronym of SYNchronous networK; at the same time a sign of *synchrony*, since Julien Pourtet was doing the dishes at the *sink* when he came up with the name.

[5] The presence of audio input from the other subject occurring during voice overlaps is a thorny issue for voice analysis. The best technical solution is to separate the speakers by placing them into adjoining studios connected by glass panes; however such a procedure adds an undesirable social inconvenience and extra distance that may well reflect on the subjects' social performance. But if the current proposed noise-cancelling approach proves inadequate, the option of physical separation may have to be discussed.

we chose the equipment to be as inexpensive as possible (no time for extra funding requests). This makes it possible to acquire the missing equipment on current laboratory funds: The webcams and microphones can be acquired for about 200 Euros, and the computers and video cameras are expected to exist in the participants' institutions.

3. ***Maximum compatibility for all members of the multi-centre study.*** COST 2102 participants from nine laboratories have indicated their potential interest in participating in the study (Lausanne [CH], Berne [CH], Salerno [I], Twente/Enschede [NL], Bielefeld [D], Budapest, Mataró [E], Limerick [IRL], Bilbao [E]). It is crucial that the two audio and video sources for detailed analysis (the webcams and the microphones) be of *exactly the same construction* and *be recorded in exactly the same manner*. Otherwise, the detailed analyses run a high risk of not being comparable, because of different levels of noise, camera or microphone resolution and recording scenario. For reasons of expense, no specific cameras are suggested for the video recordings, but current PAL interleaved or progressive standards are encouraged (rather than the NTSC standard which involves different pixel sizes).

2.2. Technical challenges and potential problems with the experimental setup

The technical setup introduces a number of technical challenges that were examined:

1. *"Analyzeability":* Can we perform all the necessary analyses with our usual analysis tools for examinations in both video and audio? The report by Keller (26.4.2007) has shown that this is possible with the proposed instruments. This has been confirmed in recent pilot projects.
2. *Recording Capacity:* Can the setup record all the information sufficiently rapidly?
3. *Viability:* Can the setup support the conditions foreseen for the experiment?
4. *Instrument integration:* Six recording instruments (two microphones and four cameras) are expected to record correctly, with high precision and synchronously. Is synchronous recording possible with sufficient resolution in both video and audio domains?

Points 2. – 4. were examined in the following pilot experiments.

2.3. Experiments on the recording quality

To examine the recording quality that is possible with this setup, two small experiments were run. The equipment was as follows:

Computer 1: Lenovo ThinkPad T60, dual-processor, 2.15 GHz, Pentium III, Logitech QuickCam STX (640x480 px), USB connection, Plantronics DSP 400 microphone

Computer 2: Lenovo ThinkPad T43p, single-processor, 2.13 GHz, Pentium M, Logitech QuickCam STX (640x480 px), USB connection, Plantronics DSP 400 microphone

Video Camera: Sony DCR-PC119E PAL (320x240 px), MiniDV cassette, Firewire connection, boom microphone

The drivers delivered with the Logitech STX web cameras only record in Microsoft's proprietary WMV format. Since we wish to have access to the technical details of the video stream, another capturing programme was searched that would be capable of recording either in uncompressed format, or in a more generally known compressed format. Of various freeware or shareware programmes that were tried out, the most recent release of the freeware programme VirtualDub (currently v. 1.7.2) distinguished itself by an excellent supply of detailed technical information, by a low frame jitter rate during capturing, and by dropping the fewest frames during recording (none) when recording at 15 fps and at 640x480 pixels, at either uncompressed AVI or in the relatively well-documented Microsoft MPEG4 v2 AVI compressed format (compression of about 50:1 - 90:1)[6].

One of the strengths of the VirtualDub programme is that in recent versions[7] it has provided adapted video recordings for specific types of cameras. The Logitech STX camera incorporates a chip that implements native 15 fps recording at 640x480 pixels in the planar (non-interleaved) I420 format[8]. VirtualDub has a specific 15 fps I420 setting for performing capturing at that size. This native format incorporates a slight compression of 1.3:1.

2.4. Experiment 1: Technical assessment

To examine differences between uncompressed and compressed recordings, one-minute recordings were made in uncompressed and MPEG4 compressed formats. For the compression, Microsoft MPEG4 v2 format was used. The system was able to record without dropping video frames and at full 48 kHz audio, as indicated in the VirtualDub recording log (Figure 2).

Compressed vs. uncompressed: Although the uncompressed video file was fairly gigantic (440 MB uncompressed vs. 6.2 MB compressed for one minute), the recording quality appears similar. The size of the files is of importance for our experiments. In uncompressed mode, an entire experiment would make 17 GB * 4 cameras = 68 GB of video files. If the above compressed format is chosen, an entire experiment would occupy less than 1 GB of data space. It is thus crucial to perform further analyses (Experiment 2), in order to see if the compressed format satisfies a given laboratory's requirements for spatial and/or sequential resolution.

[6] Format chosen after consultation with Marcos Faundez Zanuy. This video format can be analyzed with Matlab's decoding programmes.
[7] Still called "experimental" in June 2007.
[8] http://www.fourcc.org/yuv.php says: IYUV and I420. These formats are identical to YV12 except that the U and V plane order is reversed. They comprise an NxN Y plane followed by (N/2) x (N/2) U and V planes.

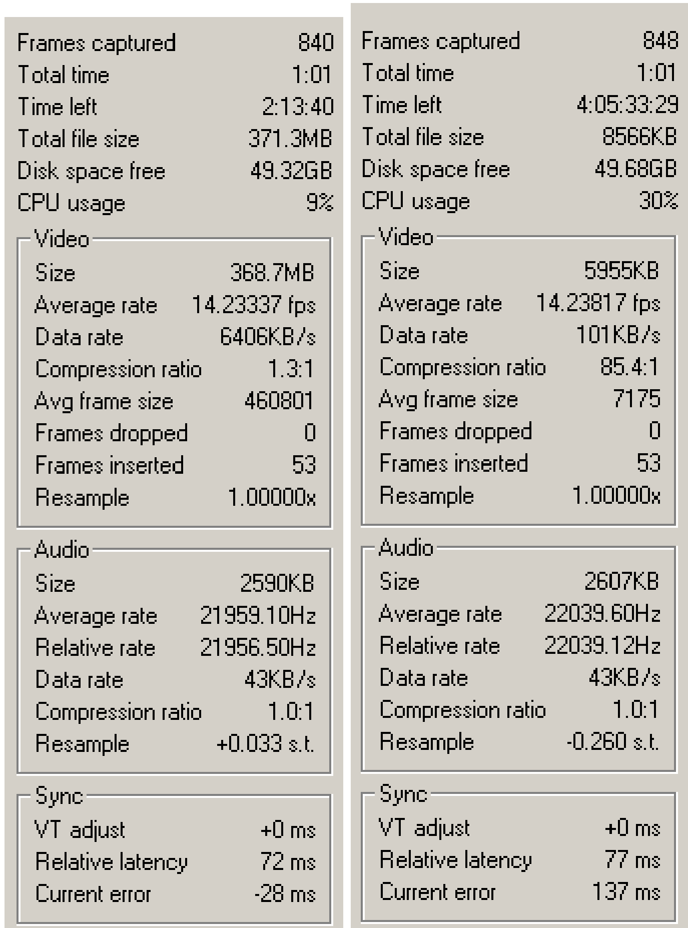


Figure 2. Technical information furnished by VirtualDub for constant bit rate uncompressed (left) and variable bit rate Microsoft MPEG4 v2 – compressed (right) recording of one minute. It can be seen that values are largely comparable, and that the main difference lies in file size. *N.B.* The compression ratio for the "uncompressed" option is given as 1.3:1, which reflects nothing more than the standard I420 compression scheme applied inevitably by VirtualDub when recording from the I420 chip in the Logitech web cam in uncompressed mode.

2.5. Experiment 2: Synchronization

Two subjects were seated in front of two portable computers, each with a Logitech web camera and with a Plantronics microphone. One subject was also filmed with a video camera. Two sets of data were recorded, uncompressed and compressed. The audio was always set at 48 KHz mono.

1. The subjects were asked to engage in free conversation for one minute with and without compression.
2. The subjects were asked to produce repeatedly /pa/ for one minute with compression. Every ten seconds, the speaker said /papapa/ to be able to compare the timing of (a) the two computers, (b) the computers vs. the video camera, and (c) the audio vs. the video.

The recordings were analyzed with Adobe's Premiere, with VirtualDub and with the ELAN Linguistic Annotator for evidence of de-synchronization. Although the microphone-camera synchronization had been unsatisfactorily adjusted in one of the two web camera setups (the voice led lip separation by about 40 ms, see section 3.2.), no deteriorating de-synchronization problem was noted. The relationship between voice onset and lip opening remained stable throughout all three recordings, and the onsets and ends of the three recordings appeared synchronous to the eye and to the ear.

3. The Experiment

3.0. The Experimental Setup

The experimental setup should strike a favourable balance between the naturalness of the social experiment and the physical and informational requirements of the experiment. The basic experiment was chosen on the basis of experience cumulated in Wolfgang Tschacher's Bern laboratory for research on psychotherapy. It was further refined in discussions with Anton Nijholt's research group at the University of Twente on 30.5.2007, and reflects considerations incorporated in an extended interview with Dirk K.J. Heylen.

We initially assumed that the conversational game would be based on the well-known and theoretically well-examined Iterative Prisoner's Dilemma game. However, some problems have arisen in the application of game (see below), and some further time must be given to precise definition of the conversational game or interaction to be recorded.

The basic experiment will be set up as follows:

1. Initially, general information about the subject and a personal assessment of the subject is taken.
2. Two subjects are seated opposite each other, in front of a computer. The distance between the two subjects is about two meters (the distance formed by the depth of the two computers).
3. Each subject is wearing a microphone around his neck. High-resolution web cameras fixed on top of the computer screens record facial gestures. Two digital cameras perform waist-up recordings over the opponent's heads.
4. The subjects are given the following instructions:
    a. *The reason for running the experiment:* The subjects are told that the data are obtained in a European project to find out how they play a cross-cultural game. The notion of a "cross-cultural experiment" is indicated so as to focus the subjects' interest away from other notions of interest in the experiment, particularly notions of agreement/cooperation, which are the veritable key variables of the experiment. The cross-cultural part of the experiment is a secondary, but less crucial aspect of the experiment.
    b. *Detailed instructions* are given of how the game is played and how long it will last.
    c. *Play emphasized*. The players are asked to play the game "as colleagues", so as to avoid excessive dominance situations.
    d. *Spontaneity*. The players are encouraged to play the game as spontaneously as they can, so as to avoid acting.

5. The subjects play two tournaments of a converstional game, one part with rewards that encourage cooperation and another with rewards that encourage competition and conflict. This order between the two parts is fixed, in order to preserve the necessary “good will” for the cooperative initial phase of the game.
6. Each tournament consists of seven phases of 3 minutes (total: 42 minutes of playing time). During the 3 minutes, subjects should try to convince the opponent of their point of view. No limits are set on the subject of discussion.
7. Subjects can see their advances or losses in the game in a bi-dimensional graph displayed on their computer screens. The current initial version of the recording programme does not yet implement this simulation.
8. After each phase of the game (after each 3 minutes), a short assessment is taken to measure how the subject is feeling about:
    a. His/her own performance
    b. The opponent’s performance
9. Subjects sign consent forms where the subjects are told they can review the video and have a portion of the video erased.
10. The subjects are reminded of the fact that this “was just a game”, in order to reverse as much as possible any ill effects of the conflictual game condition.

3.1. Recording the Experiment

An initial version of an interactive recording programme is currently available for recording (called “**Sink**”[9]). This programme consists of a server component and two client components. The server component can reside either on one of the two recording computers, or on a third computer. It has been tested and runs correctly on Windows Professional XP. It has not been tested on any other version of Windows, but there is no intrinsic reason why it should not run correctly on other versions.

The **Sink** programme currently does the following (Figure 3):

1. Each client defines a recording folder and identifies the location of the VirtualDub folder.
2. Both clients connect to the server and they indicate that they are ready to launch the experiment. When both are ready, the server signals a go-ahead to the client.
3. The two client programmes launch VirtualDub, the subjects waits a few moments (10 seconds) till the capture program has started, then the programme captures the video and the audio for the specified duration with settings that were previously specified.
4. When the capture time has been completed (typically after 3 minutes), the two client programmes close VirtualDub.
5. Future expansion of the programme is foreseen for presenting ongoing information of the game to the subjects.
6. **Sink** is open source. It was written in Borland C++.

[9] A weak take-off on “sync”, for “synchronous recording”.

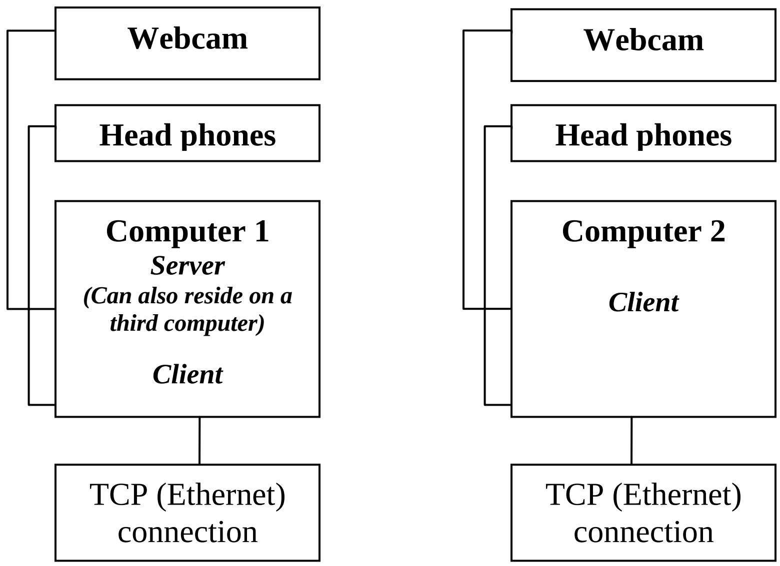


Figure 3. The server programme resides on one of the two computers and the client programme resides on both computers. Optionally, the server programme can also reside on a third computer. The computers are connected via a normal TCP connection which is preferably an Ethernet connection. If the normal Ethernet connection doesn't work, one can also establish a small local network via a switch, or via a crossed Ethernet cable.

3.2. Lip Synchronization

A pilot recording should be made with recordings of the sound /p/ (like /papapa…/) in order to set the lip synchronization offset in Virtual Dub. Put VirtualDub into File | Capture AVI mode. The offset relationship between video and audio can be changed by adjustments to the dialogue in Capture | Timing…, last section: "Number of audio blocks to use at start". We have found that the number to be provided in this box varies considerably with the specific computer-webcam combination. On one computer we set the offset to -500 blocks and on another to +200 blocks.

3.2. Detailed instructions for Recording

1. Go to Start | Control Panel | Date and Time", click on "Internet time" and time-synchronize both computers. This should be done *prior to each experiment*, because the files will be named after the recording times. During data analysis, the time information will be used to identify matching video recordings from the two subjects.
2. Launch VirtualDub and set the following settings. This needs to be done only once. The settings will be stored automatically and will be applied automatically to the capture phase:
    a. Open the Capture menus: File | Capture AVI.
    b. Device | Logitech Communicate STX web camera
    c. Video | Capture Pin | I420 | 640 x 480
    d. Video | Compression… | Microsoft MPEG4 Video Codec V2
    e. Audio | Raw Capture Format… | 22050 Hz mono 16 bits
    f. Capture | Settings… | Frame rate 15.0
    g. Capture | Timing… Set as in Figure 4

    h. Select Capture | Hide display on capture
    i. Select Capture | Show information panel
    j. Select Capture | Show status bar
3. Adjust the sound volume on both installations.
4. Download the **Sink** package from the web site noted in the references.
5. Place **SinkClient.exe** into any folder on both computers, and **SinkServer.exe** into any folder on one of the two computers, or on a third computer.
6. Identify the IP number of the server computer (Start | Run | ipconfig, and be sure to take the relevant connection [generally Ethernet]). If you have too many IP numbers active, turn the unused connections off in Control Panel | Network Connections.
7. In **Sink** with the "Connection" tab showing, enter the server IP number into the Server IP number slots.
8. Change to the "Configuration" tab and set the folders for the video recordings and for the VirtualDub programme.
9. In the menu File, activate "Save Configuration". The programme will reopen with the same configuration the next time.
10. Both subjects indicate they are ready to be recorded. When a "go-ahead" signal is received from both computers, the server sends signals to both clients to start recording.
11. At the end of the recording, signals are sent to the client programme to stop the VirtualDub recording and to close the VirtualDub programme.
12. The completed recordings are found in the respective recording folders.

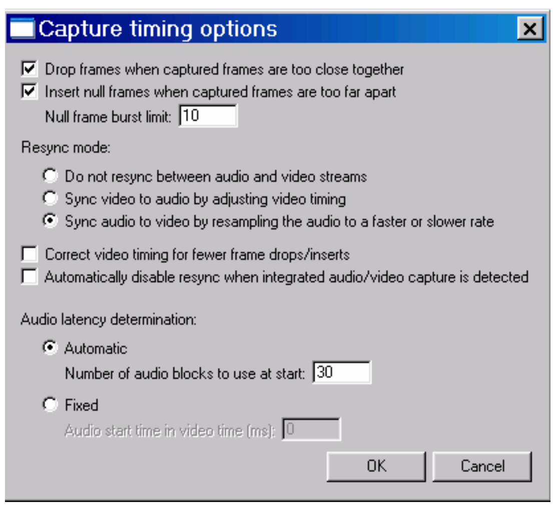


Figure 4. Initial timing options.

3.3. The Prisoner's Dilemma Game

Simulation. The prisoner's dilemma game[10] was initially chosen as the conversational game to be recorded in this experiment. The game involves a choice between cooperative

[10] See Appendix 1 for a quick summary of the basic game, and Brehm [1996] for a summary of the social effects and some of the limits of the game. A website for resources on the game is at

and selfish behaviour. In short, the players are told that they have committed a crime and have been arrested. Prior to committing the crime, they had made a pact not to tell the police anything about what either of them had done. The police offer the prisoners a deal: if they hold to the pact, they are convicted of a minor crime, but if they "snitch" on the other prisoner, the "snitcher" gets off free and the other prisoner gets convicted of a major crime. The game is played iteratively, and players can either defect from the pact or hold to the pact at every turn.

The game has been tested widely, and the theoretical aspects of the game are well explored. We implemented the game as a simulation and attempted to run a real-world pilot test, but we ran into some difficulties with the real-world application of the game.

In our simulation of more or less cooperative players working in game payoff situations that favour greater or lesser cooperation, we confirmed the basic findings of the field. When the game is played in the usual mode with common payoffs, persons defecting from the cooperative pact tend to win. When the simulation payoffs were changed to favouring cooperation, the cooperative players tended to win.

Payoffs. The usual uncooperative payoffs are as follows. They are taken from Appendix 2, and row player payoffs are given first. The abbreviations follow common use in the literature: T = temptation payoff for defecting while the other person cooperates, R = reward for cooperative behaviour, P = punishment for defecting at the same time as one's colleague, S = sucker's score for being cooperative when the other person defects:

UNCOOPERATIVE	**Cooperate**	**Defect**
Cooperate	*R = 3* *R = 3*	*S = 0* *T = 5*
Defect	*T = 5* *S = 0*	*P = 1* *P = 1*

It is noted that the payoff for defecting gives the player 5 points and gives the opposite player 0 points. This strong reward for defecting is responsible for the statistically strong success of uncooperative behaviour in this game. To create a cooperative environment, a modification was introduced on the variables T and S. Through the inversion of the rewards for defecting, the defecting player gets 0 points and the "wronged" player gets 5 points. The rewards schedule looks like this for the cooperative condition:

---

http://www2.lifl.fr/IPD/ipd.html.en. To understand the game, one can try various online versions at sites such as http://serendip.brynmawr.edu/bb/pd.html, http://www.gametheory.net/Web/PDilemma/, or http://www.open2.net/trust/dilemma/dilemma1.htm.

COOPERATIVE	**Cooperate**	**Defect**
Cooperate	*R = 3* *R = 3*	*S = 5* *T = 0*
Defect	*T = 0* *S = 5*	*P = 1* *P = 1*

For the simulations to verify the two payoff schedules, half of the simulated players took more cooperative decisions, and half took more uncooperative decisions. Under the uncooperative schedule, the uncooperative players won systematically, while under the cooperative schedule, the cooperative players won systematically.

Real-subject pilot. In preparation of the real-world application of the game, the instructions to the players were modified to account for the new possibility that defection would not necessarily pay off. Instead of simply converting the police offer of a reduced sentence into points for the player, a second decision layer was introduced. Since judges have the final word on reducing a prison term, players were instructed to consider that even though the police may promise better terms for confessing to the crime, judges may think otherwise, and players may actually end up losing points for defecting from their pact with their colleague. For the first part of the game, we then have the judges apply the cooperative playoffs, and for in the second part of the game, we have the judges apply the uncooperative playoffs.

The game was run in a pilot recording with I. Pante and J. Pourtet at the University of Lausanne according to the new instructions, but some major problems were noted. First, the present instructions did not induce the subjects to converse freely, second, the semantics of the game were ill-defined for the cooperative condition, and third, the introduction of a further decision layer left subjects unclear about what they were to do in the game. Specifically:

- *Conversational richness*. The game has to be further modified to induce players to converse and to attempt to influence each other. Considerable detail close to life has to be given to the players, so that true conversation can emerge and that social contingencies develop. Without rich interactional details, the conversations become dry and terse.
- *Semantics*. There are some problems with bending the semantics of the Prisoner's Dilemma game to a cooperative situation. Curiously, this problem is not addressed in the relevant literature. The prisoners are supposed to have made a pact not to tell the police or the judges about the crime. Motivating a prisoner's defection from this pact with a reduced prison term makes good sense in the uncooperative manner of playing the game. But how is the cooperative mode to be motivated? Logically, one would have to encourage players to hold to the pact. Yet although it is likely that police or judges think favourably of cooperation in general, it is not so sure that in real life, they would also take a positive view of the prisoner's pact to divulge nothing of their crime. In fact, it turns out to be quite difficult to bend the semantics of the prisoner's dilemma game into the role of favouring cooperation. Since the notion of simply

inverting the payoffs takes the simulation results into the right direction, it might be a better idea to invent new semantics for the cooperation-favouring part of the game.

- *Clarity about the payoffs*. The need to have clearly defined payoffs is related to the introduction of the cooperative mode of playing the game. The cooperative mode with the extra layer of judges rendered the dilemma unclear, which in turn rendered the game unplayable.

Altogether, it is clear that the conversational interaction to be asked of participants still merits some considerable further reflection.

4. The Analysis

4.0. The Mark-Up

All recordings have to be marked up at two levels:

- At the first *concrete* level, the recordings are segmented into parts relating to "agreement" / "disagreement" / "other"
- At the second *tacit* level, the recordings are segmented into "cooperative" / "uncooperative" / "other".

The *Elan Linguistic Annotator* programme is optimally suited to the annotation of the recordings of this programme[11]. It is compatible with the MPEG and AVI formats and it permits to analyze up to four video streams in parallel. The various time offsets of the different recordings can be compensated, so that all four streams show the same point in time. The programme is Java-based and is multi-platform.

5. References and Web Sites

Entry procedure for the COST 2102 Website:

Log into www.cost2102.eu
Your login data is:
username: xxx (supplied by email)
password: yyy (supplied by email)
After login, click on Tikiwiki WG1 (left menu)
Login once more with the same username/password
Click on "List galleries", left bottom of menu
Click on WG1 Files
Download the desired files

Sink programme: Use the entry procedure for COST 2102 Website and download either Sink-exe or Sink-src. Sink-src includes the executables.

Extracts from the pilot recordings: Use the entry procedure for COST 2102 Website and download 10 seconds of the /papa/ pilot project: one video file of a waist-up shot, and two web cam files, in Microsoft MPEG4 v2.

VirtualDub: freeware, needed for capturing video and audio. http://virtualdub.sourceforge.net/. Current and recommended version: 1.7.2.

WaveSurfer: freeware, capable of analyzing audio in conjunction with video. http://www.speech.kth.se/wavesurfer/

Praat: freeware, contains a large set of speech analysis tools. http://www.praat.org

Elan Linguistic Annotator: freeware, for annotating conversational passages with video and audio. http://www.lat-mpi.eu/tools/elan/

---

[11] Thanks to Dirk Heylen (Twente U.) for the suggestion of this programme.

Repository for the Prisoner's Dilemma game: http://www2.lifl.fr/IPD/ipd.html.en#jprison

Keller, E., & Tschacher, W. (in press). Prosodic and gestural expression of interactional agreement.
Heylighen, F. (2000): *The Prisoners' Dilemma*, in: F. Heylighen, C. Joslyn and V. Turchin (editors): *Principia Cybernetica Web* (Principia Cybernetica, Brussels), URL: http://cleamc11.vub.ac.be/PRISDIL.html
Brembs, B. (1996). Chaos, cheating and cooperation: potential solutions to the Prisoner's Dilemma. OIKOS 76, 14-24. URL: http://brembs.net/ipd/ipd.html.
Keller, E. (20.4.2007). Recording Setup for the Agreement-Support Audio-visual Research Project. Distributed to interested participants.
Keller, E. (26.4.2007). Recording Setup for the Agreement-Support Audio-visual Research Project: 2. An Experiment. Distributed to interested participants.

Appendix 1: A Short Description of the Prisoner's Dilemma

Reprinted from Heylighen (2000)

Cooperation is usually analysed in game theory by means of a non-zero-sum game called the "Prisoner's Dilemma" (Axelrod, 1984). The two players in the game can choose between two moves, either "cooperate" or "defect". The idea is that each player gains when both cooperate, but if only one of them cooperates, the other one, who defects, will gain more. If both defect, both lose (or gain very little) but not as much as the "cheated" co-operator whose cooperation is not returned. The whole game situation and its different outcomes can be summarized by table 1, where hypothetical "points" are given as an example of how the differences in result might be quantified.

Action of A/Action of B	**Cooperate**	**Defect**
Cooperate	Fairly good [+ 5]	Bad [- 10]
Defect	Good [+ 10]	Mediocre [0]

Table 1: outcomes for actor A (in words, and in hypothetical "points") depending on the combination of A's action and B's action, in the "prisoner's dilemma" game situation. A similar scheme applies to the outcomes for B.

The game got its name from the following hypothetical situation: imagine two criminals arrested under the suspicion of having committed a crime together. However, the police does not have sufficient proof in order to have them convicted. The two prisoners are isolated from each other, and the police visit each of them and offer a deal: the one who offers evidence against the other one will be freed. If none of them accepts the offer, they are in fact cooperating against the police, and both of them will get only a small punishment because of lack of proof. They both gain. However, if one of them betrays the other one, by confessing to the police, the defector will gain more, since he is freed; the one who remained silent, on the other hand, will receive the full punishment, since he did not help the police, and there is sufficient proof. If both betray, both will be punished, but less severely than if they had refused to talk. The dilemma resides in the fact that each prisoner has a choice between only two options, but cannot make a good decision without knowing what the other one will do.

Such a distribution of losses and gains seems natural for many situations, since the co-operator whose action is not returned will lose resources to the defector, without either of them being able to collect the additional gain coming from the "synergy" of their cooperation. For simplicity we might consider the Prisoner's dilemma as zero-sum insofar as there is no mutual cooperation: either each gets 0 when both defect, or when one of them cooperates, the defector gets + 10, and the co-operator - 10, in total 0. On the other hand, if both cooperate the resulting synergy creates an additional gain that makes the sum positive: each of them gets 5, in total 10.

The gain for mutual cooperation (5) in the prisoner's dilemma is kept smaller than the gain for one-sided defection (10), so that there would always be a "temptation" to defect. This assumption is not generally valid. For example, it is easy to imagine that two wolves together would be able to kill an animal that is more than twice as large as the largest one each of them might have killed on his own. Even if an altruistic wolf would kill a rabbit and give it to another wolf, and the other wolf would do nothing in return, the selfish wolf would still have less to eat than if he had helped his companion to kill a deer. Yet we will assume that the synergistic effect is smaller than the gains made by defection (i.e. letting someone help you without doing anything in return).

This is realistic if we take into account the fact that the synergy usually only gets its full power after a long term process of mutual cooperation (hunting a deer is a quite time-consuming and complicated business). The prisoner's dilemma is meant to study short term decision-making where the actors do not have any specific expectations about future interactions or collaborations (as is the case in the original situation of the jailed criminals). This is the normal situation during blind-variation-and-selective-retention evolution. Long term cooperations can only evolve after short term ones have been selected: evolution is cumulative, adding small improvements upon small improvements, but without blindly making major jumps.

The problem with the prisoner's dilemma is that if both decision-makers were purely rational, they would never cooperate. Indeed, rational decision-making means that you make the decision which is best for you whatever the other actor chooses. Suppose the other one would defect, then it is rational to defect yourself: you won't gain anything, but if you do not defect you will be stuck with a -10 loss. Suppose the other one would cooperate, then you will gain anyway, but you will gain more if you do not cooperate, so here too the rational choice is to defect. The problem is that if both actors are rational, both will decide to defect, and none of them will gain anything. However, if both would "irrationally" decide to cooperate, both would gain 5 points. This seeming paradox can be formulated more explicitly through the principle of suboptimization.

See also:

- an interactive implementation of the Prisoner's dilemma where you can play the game yourself
- Bjoern Brembs' review on the iterated Prisoner's Dilemma:

- Heylighen F. (1992) : "Evolution, Selfishness and Cooperation", Journal of Ideas, Vol 2, # 4, pp 70-76.

Appendix 2: The Classical Iterated Prisoner's Dilemma

Adapted from http://www2.lifl.fr/IPD/ipd.html.en

Let two artificial rational agents have the choice between two moves:

- *Cooperation*, let us note **C**
- *Defection*, let us note **D**

They play one against the other, in a synchronous manner, so that they do not know what the other will play. They get a score according to the situation of the move:

- If both cooperate, then both get the *Reward for cooperation* payoff, evaluated to *R* points;
- If both defect, then both get *self-punishment* payoff, let it be *P* points;
- If one chooses to defect while the other chooses to cooperate, then the one who has defected gets the *Temptation* payoff, let it be *T* points, and the one who has cooperated gets the *Sucker's* score, let it be *S* points.

The classical choice of value for payoff is (row player payoffs are given first):

	Cooperate	**Defect**
Cooperate	$R = 3$ $R = 3$	$S = 0$ $T = 5$
Defect	$T = 5$ $S = 0$	$P = 1$ $P = 1$

To have a dilemma, temptation must pay more than cooperation, which must pay more than punishment, which must be better than to be the sucker. This is formalised by:

$$T > R > P > S$$

Since the one-shot version of the Prisoner's Dilemma is not very interesting (the most rational choice is to defect), the game is repeated an unknown number of times. The game is said *iterated*. The final score of a payer is the sum of all its moves score. Since no player knows when the game will be ended, it is possible to study each agent's strategy, to look, for instance, how each player tries to apply cooperation in the game.

In order to favour cooperation, i.e. common interest at the expense of selfish interest, this inequality is respected:

$$2R > T + S$$

With this restriction, strategies have no advantage in alternatively cooperating and defecting. To study the behaviour of strategies, two types of computation can be performed.

- The first one is a simple **round-robin tournament**, in which each strategy encounters all other strategies. Its final score is the sum of all scores of all confrontations. At the end, the strategy's strength is given by its rank in the tournament.
- The second one is a simulated **ecological evolution**, in which at the beginning there is a fixed population including the same quantity of each strategy. A round-robin tournament is made and then the population of bad strategies is decreased whereas good strategies obtain new elements. The simulation is repeated until the population has been stabilised (the population does not change anymore). In this way, malicious strategies, those who take the initiative of the first defection, have been discovered to be not very stable, since they are invaded by kind strategies.