**A. Plan de travail pratique (3 parties)**

Recevoir et analyser BML

BAP

Valeur des joints

Répertoire des gestes

Spécification de BML (définir DTD)

Editeur des gestes pour le robot

Quoc-Anh

Quoc-Anh

Quoc-Anh + André-Marie

VC++

Java

VC++

XML

BML

Java

1-BML

2-Editeur

3 - Réalisation BML

List des keyframes

BML

**Couche intermédiaire**

Vidéos des conteurs

Symbolique



Virtual agent

Interpretation

Divers(colors)

Blending/Fusion

Body

Head Gestures

Hand Gestures

**B. En détail**

**Related work:**

**REA and BEAT use predict timing of gesture animations from synthesized speech such that the expressive phases coincides with the most prominent syllable in speech. (du doan truoc thoi gian tuyet doi cho cac gestures dua vao thoi gian cua speech).Tuy nhien no ko tinh den can bao nhieu thoi gian de thuc hien gesture (The Behavior Scheduler makes sure the stroke of the gesture aligns with the correct word, but does not attempt to stretch out the rest of the gesture, for instance to span a whole phrase that needs to be illustrated. Similary, it does not attempt to slow down or pause speech to accommodate a complex gesture, a phenomenon observed in people). => Phuong phap nay ko tao ra su linh dong (dynamic) neu phai thay doi thoi gian khi thuc hien gesures.**

**Trong he thong Max, khi biet duoc thoi gian tuyet doi cua gesture.stroke la t, Max dua ra thoi gian default bat dau gesture la t-0.3s de co the dua tay den vi tri thuc hien stroke. Gesture.hold se keo dai het affiliate truoc khi tro ve trang thai dau (The gesture stroke is either set to precede the affiliate's onset by a given offset (per default one syllable's approximate duration 0.3s) or to start exactly at the nucleus if a narrow focus has been defined. In any case, the stroke is set to span the whole affiliate before retraction starts - hold phase). Cai nay cung dc thuc hien o REA va BEAT.**

**Doi voi truong hop 2 gestures lien tiep, theo max thi stroke cua gesture 1 luon phai hoan thanh roi moi den gesture 2 - giai doan pending tren hinh minh hoa (mac du co the bo qua cac buoc nhu retraction cua gesture 1) (Simulating these mechanisms is highly context-dependent for it has to take into account properties of the subsequent stroke (form, location, timing constraints) as well as current movement conditions when the previous chunk can be RELIEVED, i.e. when its intonation phrase and its gesture stroke are completed.**

**Co the gia su bang cach luon co dinh bat dau mot gesture mot khoang thoi gian 0.3s truoc thoi gian thuc hien stroke. nhu vay, thoi gian hold se duoc keo dai phu thuoc vao khoang thoi gian thuc hien prepartion (the vocal pause between subsequent intonation phrases may thus be stretched, depending on the time consumption of the preparation phase).**

**1. Spécification de BML**

- Méthode:

+ Basé sur le bml.DTD actuel, la spécification standard (le site SAIBA) et MURML (Kopp)

+ Définir nous-mêmes les significations correspondant aux valeurs des attributs (trajectoire, direction du mouvement, forme de la main,...)

+ Le geste est décrit par une seule phase STROKE avec la direction du mouvement. Donc les autres phases seraient automatiquement créées par le système. Exemple:

<bml>

 <gesture>

<id> small </id>

<type> ICONIC </type>

<hand> BOTH </hand>

<lexeme> </lexeme>

<handshape> OPENHAND </handshape>

<extendedfingerdirection> AWAYUP </extendedfingerdirection>

<palmdirection> INWARDS </palmdirection>

<trajectory> straight </trajectory>

<twohanded> mirror </twohanded>

<location\_vertical> CENTER </location\_vertical>

<location\_horizontal> CENTER </location\_horizontal>

<location\_distance> NEAR </location\_distance>

<movementdirection> INWARDS </movementdirection>

**<timeminimum> 0.5 </timeminimum>**

</gesture>

</bml>

+ Pour les gestes compliqués, on va référer à une animation élaborée par Chorégraphe en utilisant l'attribut "lexeme".

<bml>

<gesture>

<id> pear </id>

<type> LEXICALIZED </type>

<lexeme> behavior.xar </lexeme>

</gesture>

</bml>

**2. Moteur de contrôler les gestes (head gestures and hand gestures)**

2.1. Feedback

Objectif: Pour modifier et adapter les contraints du robot (temps de la réalisation et forme du geste).

API:

Description du geste avec BML

GesturePlanner

TurnPlanner

Keyframes

MotorPlanner

API

Chu y cai o <precedent mouvement a fini> phai chia lam may kieu: stroke finit? neu da ket thuc thi xet tiep den viec phan bo tai nguyen <arm est disponible?>, roi tiep tuc den viec tinh toan du bao thoi gian de thuc hien gesture (phu thuoc vao hinh dang, duong di,...) hinh nhu cai nay can tien hanh truoc khi phan bo tai nguyen de biet duoc can thuc hien truoc stroke bao lau va den thoi diem do thi nhung tai nguyen can thiet co san dung ko.

Nen nho la vi Head va Gesture doc lap nhau, nen co the tien hanh vao 2 modul rieng biet.

Notes:

1. Abstract Gesture with BML has only descriptions of stroke phase (and trajectory information)

2. Gesture Editor uses the same gesture engine to visualize the elaborated results

3. For each entry in the repertoire

<bml>

<gesture>...</gesture>

<head>..</head>

<torso>..</torso>

<gaze>..</gaze>

</bml>

Ok from Catherine but AM (have to modify many things)

Specification of BML

Receive BML through Psyclone and initiate signals (speech, gesture, head,..)

GESTURE EDITOR

ACAPELA

Send speech signal to TTS and receive the absolute times (time markers)

Repertoire of gestures

Define intermediate wrist positions based on tags "direction" and "trajectory"

Load descriptions of gestures from the repertoire (i.e. stroke phases)

Visualisation

Receive the feedback from the robot about the execution of precedent gesture

Receive the synchronized time from the central clock

Convert gesture abstracts to values of articulations with parameters (space,...)

Repertoire of predefined key gestures with articulation values

Non

gesture[i].start\_time >= syn\_time?

Yes

gesture[i-1].stroke\_end finished?

check resources through API areResourceAvailable(name)?

Non

Non

Non

fix the position by API setPosition(gesture[i].end\_position)

free necessary resources through API killTasksUsingResources(name)

gesture[i].end\_position==getPosition()?

check if it finishes through API isRunning(id) or overtime?

Non

Yes

execute the gesture.stroke through API (id=positionIIterpolation)

Schedule

i=i+1

Yes

BML

**Gesture Engine for the robot**

stroke start

stroke end

stroke end

GESTURE [i]

relax

relax

stroke start

.......

GESTURE [i-1]

time

relax

relax

end\_time

start\_time

Notes:

1. Stroke phase lasts from stroke start to stroke end. It includes perhaps many keyframes that form the trajectory of gesture following the BML description.

2. The algorithm does not compute all gestures before sending them as usual but it sends each computed gesture in turn in order to give feedback and adjust info.

3.

move to relax position by API positionInterpolation(relax)

Non

Yes

gesturel[i].end\_time >= syn\_time?

Receive the synchronized time from the central clock

No

No

\* TurnPlanner: Traiter le BML recu et charger les gestes correspondants à partir du répertoire des gestes.

\* GesturePlanner: Faire planning et élaborer les keyframes

\* MotorPlanner: Calculer la valeurs des articulations suivant le keyframe et planner les temps de finir ce keyframe. Il devrai recevoir en cours les feedback pour assurer que l'animation sera bien finie à temps et à correcte position.

Khong the thuc hien lop trung gian boi vi duong di cua geste cho Robot se khac so voi Greta. Nhu vay o day khong han la keyframe tren duong di, ma chi la keyframe cu the hoa cho tung geste.

Neu muon xu ly feedback thi khong the doi den luc tinh toan het roi gui toan bo den robot ma phai thuc hien online vua tinh toan vua gui. Moi khi nhan duoc feedback phai tinh toan lai va gui tiep.

<Keyframe>

<Start\_Time>3</Start\_time>

<Head> <Direction\_Dynamic>Left\_Hand:Direction</Head>

<Gesture> </Gesture>

<Keyframe>

Notes: Il devrait mieux plutot utiliser le coordonnee decarter XYZ que l'espace des joints. Car on pourra facilement de faire une comparaison entre deux agents par x(yz)-coordinate diagramme.

1) vector<float> getPosition (const string& name, const int& space, const bool& useSensorValues) \*

2) bool killTask (const int& motionTaskID)

3) bool isRunning (const int& id)

4) bool isRunning (const string& name)

5) void stop (const int& brokerTaskID)

# Example showing how to cancel a task.

# When starting a task with 'post' you receive a taskID

taskID = proxy.post.angleInterpolation("HeadYaw",0.2,1.0,True)

time.sleep(0.2)

# Now cancel the task

proxy.stop(taskID)

**2. Editeur des gestes pour le robot**

- Durée de travail: 2 mois

Spécification du BML

Editeurs des gestes pour Nao



ID

Valeur

Keyframes (pre, stroke\_start,stroke,stroke\_end,ret)

Valeurs des joints

Type

Valeur

..................

Contrôle des paramètres

Répertoire des gestes

- Spécification:

+ Charger et enregistrer un geste dans répertoire avec le syntaxe BML

+ Visualisation des résultats (connecté au robot virtuel de Chorégraphe, GretaPlayer,..)

\* Lorsqu'on modifie la valeur de l'attribut du geste

\* Lorsqu'on change le type de la trajectoire du mouvement <trajectory>

\* Lorsqu'on change la direction du mouvement <movementdirection>

\* Lorsqu'on modifie un paramètre <spatialité, vitesse, fluidité,.. >

+ Tester la durée minimum nécessaire pour que le geste élaboré soit réalisable </timeminimum> . Cette valeur est important dans le module de réalisation BML (Si la durée attribuée au geste est trop court, le geste ne pourrait pas être envoyé au robot)

**2.1. Algorithme (geste abstrait --> valeurs absolues des joints du robot):**

2.1.1. *Position de la main*: Prédéfinir valeurs des joints du robot correspondant aux combinaisons de trois coordonnées (location\_vertical, location\_horizontal, location\_distance). Exemple:

(location\_vertical, location\_horizontal, location\_distance) =(HIGH,OUTWARD,NEAR) =(LShoulderRoll,LElbowYaw,LElbowRoll,LWristYaw,RShoulderPitch,RShoulderRoll,RElbowYaw,RElbowRoll)=(0.06,-1.55,-1.51,0.01,0.22,-0.07,1.61,1.56,0)

+ Avantage: toutes les positions sont réalisables pour le robot

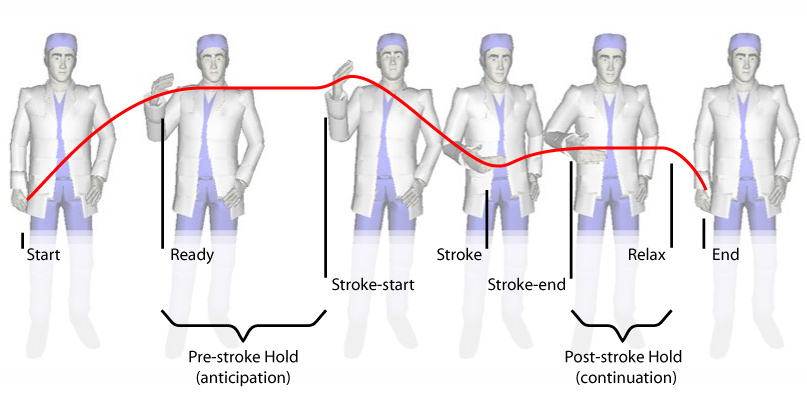
+ Désavantage: Difficile d'appliquer des paramètres tels que "spatialité", "fluidité",...

2.1.2. *Forme de la main*: Trois valeurs (ouvert, demi-ouvert, fermé)

2.1.3. *Orientation de la paume*: dépendre à la position de la main et la valeur de cet attribut. A définir ultérieur.

2.1.4. *Orientation des doigts*: Rien à faire parce que le robot n'a pas poignet dynamique. Si on change l'orientation des doigts, la position de la main changera aussi.

2.1.5. *Trajectoire*: Il faut définir nous-mêmes un/deux positions intermédiaires (STROKE\_START,STROKE\_END,...) pour avoir un trajectoire correspondant

2.1.6. *Direction du mouvement*: Pour définir les positions intermédiaires. Exemple: Si la valeur est INWARD (les mains bougent vers au centre), une position intermédiaire devrait être définie de façon qu'elle a même location\_vertical et location\_distance avec la position du stroke mais sa location\_horizontal est plus loin du centre (STROKE\_START).

**2.2. Base théorique pour élaborer le répertoire**

- Littératures: les gestes extraits des vidéos des conteurs de LIMSI et de la thèse de Calbris.

- Format: pour chaque intention, il y a certains gestes correspondants avec la durée minimale de réalisation dont le robot pourrait choisir un (dépend à la durée attribuée - la parole)

**3. Réalisation de BML**

- Durée de travail: 4 mois

3.1. Exemple pour un processus de réaliser les gestes d'un BML

<bml>

 <gesture>

<id> rhbeat </id>

<type>BEAT </type>

<hand> RIGHT</hand>

<lexeme> </lexeme>

<handshape> FIST </handshape>

<extendedfingerdirection> OUTWARDSAWAY </extendedfingerdirection>

<palmdirection> OUTWARDSTOWARDS </palmdirection>

<trajectory> straight </trajectory>

<twohanded> </twohanded>

<location\_vertical> CENTER </location\_vertical>

<location\_horizontal> CENTER </location\_horizontal>

<location\_distance> NEAR </location\_distance>

<movementdirection> DOWNOUTWARDS </movementdirection>

**<timeminimum> 0.5 </timeminimum>**

</gesture>

</bml>

<bml>

<gesture>

<id> un\_soir </id>

 <type> ICONIC </type>

<hand> BOTH </hand>

<lexeme> </lexeme>

<handshape> OPENHAND </handshape>

<extendedfingerdirection> UP </extendedfingerdirection>

<palmdirection> INWARDS </palmdirection>

<trajectory> straight </trajectory>

<twohanded> mirror </twohanded>

<location\_vertical> HIGHT </location\_vertical>

<location\_horizontal> OUTWARDS </location\_horizontal>

<location\_distance> NEAR </location\_distance>

<movementdirection> UP </movementdirection>

**<timeminimum> 0.5 </timeminimum>**

</gesture>

</bml>

<bml>

<speech start=1>

<tm id="tm1"/> Voilà *(tm1=1s)*

<tm id="tm2"/>bien *(tm2=2s)*

<tm id="tm3"/> longtemps, *(tm3=3s)*

<tm id="tm4"/> *(tm4=4s)*

<tm id="tm5"/> un soir *(tm5=4.3s)*

<tm id="tm6"/> de printemps, *(tm6=6s)*

<tm id="tm7"/> *(tm7=7s)*

<tm id="tm8"/> trois *(tm8=8s)*

speech/>

<gesture id="rhbeat" start="s1:tm1" end="s1:tm4" stroke="s1:stm3">

<description level="1" type="naobml">

<reference>beat=rhbeat</reference>

< /description>

</gesture>

<gesture id="un\_soir" start="s1:tm5" end="tm7" stroke="s1:tm6">

<description level="1" type="naobml">

<reference>iconic=un\_soir</reference>

< /description>

</gesture>

<bml/>

3.0

Stroke

geste1 (rhbeat)

Ret

Pre

1.0

0.0

4.0

Stroke\_start

6.0

Ret

7.0

6.0

keyframes

geste2 (un\_soir)

4.3

Stroke

Stroke\_start

4.0

3.0

7.0

4.3

1.0

keyframe1(1.0, stroke\_start1)

keyframe2(3.0, stroke1)

keyframe3(4.0, Retraction1)

keyframe4(4.3, stroke\_start2)

keyframe5(6.0, stroke2)

keyframe6(7.0,retraction2)

while(GetCurrentTime()==KeyFrames[i].Time)

{

If(KeyFrames[i-1].IsRunning) then KeyFrames[i-1].Stop;

KeyFrames[i].Start;

}

3.2. Notes

- Pour ne pas devoir modifier trop le module actuel, on maintient le format actuel du fichier BML comme l'entrée pour le BR (Behavior Realizer). C'est-à-dire qu'on utilise encore les tags <description> et <reference> pour lier le fichier du geste dehors et les paramètres de qualité du geste.

- Le processus d'un geste se compose d'au mois deux phases (Préparation et Stroke) et d'au maximum 5 phases (Préparation, Stroke\_start, Stroke, Stroke\_end, Rétraction) selon des cas comme suivants:

+ Les attributs <Trajectory> et <DirectionMovement> sont nuls. Alors, on n'a besoin de faire un déplacement des mains vers la position Stroke et après une rétraction à la position de repos.

+ Dans le cas la trajectoire et une ligne droite, il faut définir la position au début de la ligne (Stroke\_start) et la fin de la ligne est la position du stroke.

+ Dans le cas la trajectoire est compliqué (cercle, ziczac,..) il faut définir la position de la phase Stroke\_start et aussi Stroke\_end.

- Assumer que les phase Préparation et Retraction sont identiques et elles correspondent à la pose de repos. Le temps de commencer un geste devrait être plus tôt de temps du stroke (minimum 1s) car le stroke du geste devrait toujours se passe à la fois avec la phrase soulignée. Par exemple dans le cas ci-dessus, le geste2 démarre lorsque le geste1 n'a pas encore fini. Donc il faut arrêter le geste1 avant de commencer le geste2 pour que les mains soient placées à la position du stroke à temps (ainsi dans le cas un geste réfère à une animation prédéfinie).

3.2. Traitement des fichiers BML envoyés tour à tour

- Situations:

+ Feedback du robot (changer intention, tomber,...)

+ La parole est trop longue. Elle est donc coupé en phrases séparées et chaque phrase correspond à un fichier BML envoyé au module de réalisation.

- Algorithme:

+ Chaque BML possède un temps global (vient de horloge centrale). Les time markers sont les temps locaux et ils devraient être ajoutés par le temps global pour devenir un valeur absolu. Par exemple pour le fichier BML ci-dessus, s'il est attribué un temps global 12s, les time markers seront (tm1,tm2,...)=(13s,14s,...)

+ Lorsque le système finit de calculer le BML actuel, il va arrêter complètement l'exécution du BML précédent pour réaliser le fichier BML actuel.

**4. Couche intermédiaire**

- inspiré par EMBR (Alexis Heloir et Michael Kipp)

- Se compose des keyframes dans lesquels chaque keyframe a un temps absolu et l'état du geste à ce moment-là

**5. Divers**

- Utiliser l'éditeur des gestes pour Greta pour élaborer son propre répertoire des gestes. Il y aurait un mapping entre deux répertoies (du robot et du Greta).

- Les mapping entre FML et BML sont étendus pour couvrir les répertoires des gestes. C'est-à-dire qu'il faut élaborer des scripts pour lexicon\_nao.xml et lexicon\_greta.xml. Les deux lexicons détermineront les gestes compatibles pour chaque agent. Exemple:

<behaviorset name="emotion-panic\_fear">

<signals>

.......

<signal id="1" name="emotion=panic\_fear\_signal\_nao" modality="gesture"/> <--pour NAO--->

[ <signal id="1" name="emotion=panic\_fear\_signal\_greta" modality="gesture"/> <--pour GRETA--->]

</signals>

</behaviorset>

- BML créé par le système actuel a les temps absolus au lieu des temps relatives. C'est un problème lorsqu'on voudrait utiliser un autre TTS pour le module BR parce que le changement de la durée de la parole amène le changement des temps de synchronisation des nonverbaux => modifier le module de Planification des comportement (FML->BML)

- Créer des exemples complets pour tester et évaluer.

C. Plan de travail et calendrier d'implementation