

ຄໍານຳ

ພ້ອມນີ້ພວກຂ້າພະເຈົ້າຂໍສະແດງຄວາມຂອບໃຈ ແລະ ຮູ່ບຸນຄຸນຢ່າງສູງມາຍັງຄະນະນຳ ຂອງມະຫາວິທະຍາໄລແຫ່ງຊາດ, ຄະນະກຳມະການວິຊາການ, ຄະນະບໍດີຄະນະວິທະຍາສາດທຳມະຊາດ, ຄະນະພາກວິຊາວິທະຍາສາດຄອມພິວເຕີ ແລະ ບໍ່ກ່ຽວຂ້າງໜີ້ສະໜັບສະໜັນການພິມປຶ້ມເຖິ່ງນີ້.

ນະຄອນຫຼວງວຽງຈັນ, ວັນທີ 16 ສິງຫາ 2018

ຄະນະຮູບຮູງ ແລະ ກວດແກ້

ສາລະບານ

ບົດທີ 1 ຄວາມຮູ້ພື້ນຖານກ່ຽວກັບປັນຍາປະດິດ (Introduction to Artificial Intelligence).....	1
1.1 ປັນຍາປະດິດແມ່ນຫຍັງ?	1
1. ລະບົບທີ່ຄືດຄືມະນຸດ (Systems that think like humans)	1
2. ລະບົບທີ່ເຮັດວຽກຄືມະນຸດ (Systems that act like humans)	1
3. ລະບົບທີ່ຄືດຢ່າງມີເຫດຜົນ (Systems that think rationally).....	2
4. ລະບົບທີ່ເຮັດວຽກຢ່າງມີເຫດຜົນ (Systems that act rationally).....	2
1.2 ປະຫວັດຄວາມເປັນມາຂອງປັນຍາປະດິດ (History of AI)	3
1.3 ການປະຢູກໃຊ້ AI ໃນປັດຈຸບັນ	5
1.4 ປະເພດຂອງປັນຍາປະດິດ (AI Category)	7
1.5 ປັນຍາປະດິດ ແລະ ບີກາດຕັ້ງ (AI and Big data)	9
ບົດທີ 2 ການແກ້ປັນຫາດ້ວຍເຕັກນິກການຄົ້ນຫາ (Problem Solving using Search)	11
2.1. ຂອບເຂດຂອງປັນຫາ (State Space)	11
2.1.1 ກຣາບ (Graph)	12
2.1.2 ເຕືອງສະຖານະຈຳກັດ (Finite State Machine)	15
2.1.3 ການກຳນົດນິຍາມໃຫ້ກັບປັນຫາ.....	17
2.1.4 ວິທີການຕ່າງໆ ໃນການແກ້ປັນຫາ	22
2.1.5 ການສະແດງຄວາມຮູ້	23
2.1.6 ຂະບວນການໃນການເລືອກກົດເກັນ	24
2.2. ການຄົ້ນຫາແບບງົມມືດ (Blind Search)	27
2.2.1 ການຄົ້ນຫາແບບລ່ວງເລີກກ່ອນ	28
2.2.2. ການຄົ້ນຫາແບບລວງກວ້າງກ່ອນ	30
2.3. ການຄົ້ນຫາແບບຮົມສະຕິກ (Heuristic Search)	32
2.3.1. ໄຕ໌ຂຶ້ນພູ (Hill Climbing)	33
2.3.2. ການຄົ້ນຫາແບບດີທີ່ສຸດກ່ອນ(Best-First Search)	36

2.3.4. ອານກຳຮິດທຶນ A*	37
2.4. MINIMAX ALGORITHM.....	42
2.5. Alpha-Beta Cutoffs	43
ບົດທີ 3 ການສະແດງຄວາມຮູ້ (knowledge Representation).....	45
3.1 Semantic Network.....	45
3.1.1. Taxonomical Hierarchy	47
3.1.2. ການຫາເຫດຜົນດ້ວຍວິທີຂອງ Semantic Network	48
3.2. Frame.....	50
3.2.1. ການຖາຍທອດຄຸນສົມບັດ.....	51
3.2.2. Slot	52
3.3. ການເພີ່ງພາດ້ານມະໂນພາບ (Conceptual dependency).....	54
3.3.1. ກິດຂອງການເພີ່ງພາດ້ານມະໂນພາບ	55
3.3.2 ກິດຕ່າງໆ ແລະ ໂຄງສ້າງຂອງ CD.....	55
3.3.3 ACT ຊະນິດຕ່າງໆໃນ CD.....	56
3.3.4 ກາລະຕ່າງໆ	57
3.3.5 ວິທີການສະແດງປະໄຫຍກທາງພາສາສາດ ດ້ວຍການເພີ່ງພາດ້ານມະໂນພາບ	57
ບົດທີ 4: ການຫາຂໍສະຫຼຸບພາຍໄຕຄວາມບໍ່ແນ່ນອນ (Inference under Uncertainty).....	59
4.1 Uncertainty.....	59
4.1.1 ການແກນຄ່າຄວາມບໍ່ແນ່ນອນ	60
4.1.2 ການຫາເຫດຜົນແບບ Monotonic	60
4.1.3 ຄ່າຄວາມແນ່ນອນ	61
4.2 ຫົດສະດີ Bayes	65
4.3 ຫົດສະດີ Dempster and Shafer	70
4.3.1. Mass Function.....	71
4.3.2 Belief Function	73

4.3.3 Plausibility Function	74
4.4 Bayesian Network	76
4.4.1 ໂຄງສ້າງຂອງ Bayesian Network	76
4.4.2 ຫິດສະດີຂອງ Bayesian Network	79
4.4.3 ການອະນຸມານຂອງ Bayesian Network	82
4.5 Fuzzy Logic.....	82
ບົດທີ 5 ແນະນຳການຮຽນຮູ້ຂອງເຕືອງຈັກ: (Introduction to Machine Learning)	87
5.1 ການຮຽນຮູ້ຂອງເຕືອງຈັກ.....	87
5.1.1 Supervised Learning	89
5.1.2 Unsupervised Learning	89
5.2 ການຕັດສິນໃຈແບບຕົ້ນໄມ້ (Decision Tree)	89
5.3. ການຮຽນຮູ້ດ້ວຍ Decision Tree.....	90
5.4. Ensemble Learning.....	91
5.5. ຕົວຢ່າງການປະຍຸກໃຊ້ Decision Tree.....	92
5.6. Nearest Neighbor Classification	100
ບົດທີ 6 ເຄືອຂ່າຍເສັນປະສາດຖາມ.....	105
6.1 ຄວາມໝາຍ	106
6.2. ອົງປະກອບ ແລະ ໂຄງສ້າງການເຮັດວຽກ	107
6.2.1. ຂໍ້ມູນນຳເຂີ້າ (Input).....	107
6.2.2. ນຳໜັກ (Weight).....	107
6.2.3. ພັງຊັ້ນການລວມ (Summation Function)	108
6.2.4. ພັງຊັ້ນການແປງ (Transformation Function).....	108
6.2.5. ຜົນຮັບ (Output).....	108
6.3. ໂຄງສ້າງ	108
6.4. ຫຼັກການ	109

6.5. ការເຮັດວຽກ.....	109
6.5.1. Back propagation Algorithm	110
6.5.2. ការຮົງຮູ້ສໍາລັບ Neural Network	112
6.6. Network Architecture.....	113
6.6.1. Feedforward network.....	113
6.6.2. Feedback network.....	113
6.6.3. Network Layer.....	114
6.6.4. Perceptrons.....	115
6.7. ປະໂຫຍດຂອງເຄືອຂ່າຍເສັ້ນປະສາດທຸກ	116
6.8 ການປະຢຸກໃຊ້ Neural Network	116
6.8.1 ການປະຢຸກໃຊ້ເຄືອຂ່າຍເສັ້ນປະສາດທຸກໃນວຽກທຸລະກິດ	117
6.8.2. ຕົວຢ່າງການເຮັດວຽກຂອງເຄືອຂ່າຍເສັ້ນປະສາດ	118
ບົດທີ 7 ລະບົບຜູ້ຊ່າງວຊານ	119
7.1. ປະຫວັດຂອງລະບົບຜູ້ຊ່າງວຊານ	119
7.2. ມີຍາມຂອງລະບົບຜູ້ຊ່າງວຊານ	120
7.3 ອົງປະກອບຂອງລະບົບຜູ້ຊ່າງວຊານ	120
7.3.1 ຖານຄວາມຮູ້ (Knowledge-Based)	121
7.3.2 ເຄື່ອງອະນຸມານ (Inference engine)	121
7.3.3 ພາສາ ແລະ ເຄື່ອງມື	129
ບົດທີ 8 ການພັດທະນາລະບົບຜູ້ຊ່າງວຊານ	131
8.1. ການພັດທະນາລະບົບຜູ້ຊ່າງວຊານ	131
8.1.1. ການຈຳແນກບັນຫາ ແລະ ວິເຕະຄວາມຮູ້ທີ່ຈະສະຫຼຸບໃສຖານຄວາມຮູ້	133
8.1.2. ການເລືອກເຄື່ອງມື ແລະ ສ້າງຄວາມເຂົ້າໃຈກ່ຽວກັບລັກສະນະຂອງການໃຫ້ຄໍາປຶກສາ	136
8.1.3. ການອອກແບບ	137
8.1.4 ການສ້າງຕົ້ນແບບ	137

8.1.5 ການຂະໜາຍາຍ, ບົດສອບ ແລະ ການປັບປຸງລະບົບ	138
8.2. ການເລືອກປັບໜາໃຫ້ເໝາະສົມສໍາລັບການພັດທະນາລະບົບຜູ້ຊ່າງວຊານຂະໜາດໃຫ້ຢ່າງ	138
8.2.1. ການຈຳແນກຊະນິດຂອງເຄື່ອງມືທີ່ໃຊ້ໃນການພັດທະນາລະບົບຜູ້ຊ່າງວຊານ	139
8.2.2. ໂຄງສ້າງຂອງເຄື່ອງມືທີ່ໃຊ້ໃນການພັດທະນາລະບົບຜູ້ຊ່າງວຊານ	139
8.3. ການໃຊ້ລະບົບຜູ້ຊ່າງວຊານສໍາລັບການວາງຜັງໂຮງງານ	142
8.3.2 ການຄົ້ນຫາແບບຮົກວິດສະຕິກ (Heuristic Search)	144
8.3.3 ລະບົບຜູ້ຊ່າງວຊານ	145

ບົດທີ 1 ຄວາມຮູ້ພື້ນຖານກ່ຽວກັບບັນຍາປະດິດ (Introduction to Artificial Intelligence)

1.1 បំណុលបច្ចុប្បន្ននៃការអនុវត្តន៍?

ມີຄໍານິຍາມຂອງບັນຍາປະດິດ (Artificial Intelligence :AI) ຫຼວງຫຼາຍ ເຊິ່ງສາມາດຈັດແບ່ງອອກເປັນ 4 ປະເພດໂດຍເປົ້າໃນ 2 ມິຕີ ໄດ້ແກ່: ລະຫວ່າງ ນິຍາມທີ່ເນັ້ນລະບົບທີ່ຮຽນແບບມະນຸດ ກັບ ນິຍາມທີ່ເນັ້ນລະບົບທີ່ເປັນລະບົບທີ່ມີເຫດຜົນ (ແຕ່ບໍ່ຈຳເປັນຕ້ອງຄືມະນຸດ), ລະຫວ່າງ ນິຍາມທີ່ເນັ້ນຄວາມຄິດເປັນຫຼັກ ກັບ ນິຍາມທີ່ເນັ້ນການເຮັດວຽກເປັນຫຼັກ

1. ລະບົບທີ່ຄິດລົມໝາດ (Systems that think like humans)

- ▶ [AI ແມ່ນ] ຄວາມພະຍາຍາມໃໝ່ອັນຫັດຕື່ນເຕັນທີ່ຈະເຮັດໃຫ້ຄອມພົວເຕີຄິດໄດ້ ... ເຄືອງຈຳກິດຈະກຳທີ່ຈະເຮັດໃຫ້ຄອມພົວເຕີຄິດໄດ້ ... ເຄືອງຈຳທີ່ມີສະຕິປັນຍາຢ່າງລົບຖວນແລະແທ້ຈິງ ("The exciting new effort to make computers think ... machines with minds, in the full and literal sense." [Haugeland, 1985])
 - ▶ [AI ແມ່ນ ກິນໄກຂອງ] ກິດຈະກຳທີ່ຈະເຮັດໃຫ້ຄອມພົວເຕີຄິດມະນຸດ, ເຊັ່ນ: ການຕັດສິນໃຈ, ການແກ້ບັນຫາ, ການຮຽນຮູ້ ("[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning." [Bellman, 1978])

2. ລະບົບທີ່ເຮັດວຽກຄືມະນຸດ (Systems that act like humans)

- ▶ [AI ແມ່ນ] ວິຊາຂອງການສ້າງເຄື່ອງຈັກລະຫີ່ເຮັດວຽກໃນສິ່ງເຊິ່ງອາໄສປັນຍາເມື່ອເຮັດວຽກໂດຍມະນຸດ (*"The art of creating machines that perform functions that requires intelligence when performed by people."* [Kurzweil, 1990])
 - ▶ [AI ແມ່ນ] ການສຶກສາວິທີເຮັດໃຫ້ຄອມພົວເຕີເຮັດວຽກໃນສິ່ງທີ່ມະນຸດເຮັດໄດ້ດີກວ່າໃນຂະນະນັ້ນ (*"The study of how to make computers do things at which, at the moment, people are better."* [Rich and Knight, 1991])

*ໝາຍເຫດ ການເຮັດວຽກຄືມະນຸດ, ເຊັ່ນ:

- ສື່ສານໄດ້ດ້ວຍພາສາທີ່ມະນຸດໃຊ້, ເຊັ່ນ: ພາສາລາວ, ພາສາອັງກິດ. ຕົວຢ່າງ: ການແປງຂໍ້ຄວາມເປັນຄໍາເວົ້າ ແລະ ການແປງຄໍາເວົ້າເປັນຂໍ້ຄວາມ.
 - ມີປະສາດຮັບສຳພັດຄ້າຍຄືມະນຸດ, ເຊັ່ນ: ຄອມພິວເຕີຮັບພາບໄດ້ໂດຍອຸປະກອນຮັບສຳພັດ ແລ້ວເອົາພາບໄປປະມວນຜົນ.
 - ເຄື່ອນໄຫວໄດ້ຄ້າຍຄືມະນຸດ, ເຊັ່ນ: ຫຼຸ່ມຍົນຊ່ວຍວຽກຕ່າງໆ. ຕົວຢ່າງ: ການດູດຜຸນ, ເຄື່ອນຍ້າຍສິ່ງຂອງ.

- ຮົງຮູ້ໄດ້ໂດຍສາມາດກວດຈັບຮູບແບບການເກີດຂອງເຫດການໃດໆ, ແລ້ວປັບຕົວສູ່ສິ່ງແວດລ້ອມທີ່ປ່ຽນໄປໄດ້.

3. ລະບົບທີ່ຄືດຢ່າງມີເຫດຜົນ (Systems that think rationally)

- ▶ [AI ແມ່ນ] ການສຶກສາຄວາມສາມາດໃນດ້ານສະຕິປັນຍາໂດຍການໃຊ້ແບບຈໍາລອງການຄິດໄລ່ ("The study of mental faculties through the use of computational model." [Charniak and McDermott, 1985])
 - ▶ [AI ແມ່ນ] ການສຶກສາວິທີການຄິດໄລ່ທີ່ສາມາດຮັບຮູ້, ໃຊ້ເຫດຜົນ ແລະ ເຮັດວຽກ ("The study of the computations that make it possible to perceive, reason, and act" [Winston, 1992])
*ໝາຍເຫດ ຄິດຢ່າງມີເຫດຜົນ ຫຼື ຄິດຖືກຕ້ອງ, ເຊັ່ນ: ໃຊ້ຫຼັກການຕັກສາດໃນການຄິດຫາຄໍາ ຕອບຢ່າງມີເຫດຜົນ, ເຊັ່ນ: ລະບົບຜູ້ຂ່າວຊານ.

4. ລະບົບທີ່ເຮັດວຽກຢ່າງມີເຫດຜົນ (Systems that act rationally)

- ▶ ບັນຍາປະດິດແມ່ນການສຶກສາເພື່ອອອກແບບຕົວເຮັດວຽກທີ່ມີບັນຍາ (“Computational Intelligence is the study of the design of intelligent agents” [Poole et al., 1998])
 - ▶ AI ກົງວຂ້ອງກັບພິດຕິກຳທີ່ສະແດງບັນຍາໃນສິ່ງທີ່ມະນຸດສ້າງຂຶ້ນ (“AI ... is concerned with intelligent behavior in artifacts” [Nilsson, 1998])

❖ Turing Test

ย้อนกลับตีนไปในปี 1950 บั๊งຈາກຄວມພິວເຕີໄດ້ຖືກຄົ້ນຄົດຂຶ້ນ Alan Turing ໄດສະເໜີການທີ່ດີສອບປັນຍາຂອງເຄື່ອງຄອມພິວເຕີ ແລະ ໄດກາຍເປັນທີ່ຮູ້ຈັກໃນນາມ ການທີ່ດີສອບທິ່ວວິ່ງ (Turing Test), ໃນການທີ່ດີສອບນີ້ ຄອມພິວເຕີ ແລະ ມະນຸດໄດ້ ສິນທະນາກັບຜູ້ທີ່ດີສອບເປັນເວລາ 5 ນາທີ ແລ້ວຈາກນັ້ນຜູ້ທີ່ດີສອບໄດ້ຕັດສິນໃຈວ່າຮັນໄດ້ແມ່ນຄົນ ອັນໄດ້ແມ່ນຄອມ. ມີການແຂ່ງຂັນປະຈຳປີເຊິ່ງເອີ້ນວ່າ the Loebner Prize, ເປັນການແຂ່ງໂປຣແກຣມທີ່ຄ້າຍຄືມະນຸດ, ສໍາລັບໂປຣແກຣມທີ່ເຂົ້າແຂງຂັນ ບໍ່ມີໂປຣແກຣມໄດ້ທີ່ເຮັດໄດ້ເຫຼົາກັບ Turing Predicted. ແຕ່ພວກເຂົາກໍເຮັດໄດ້ດີຂຶ້ນຫຼຸກຄັ້ງ ເຊັ່ນດູວກັນກັບ ໂປຣແກຣມໝາກລຸກທີ່ສາມາດຊະນະ grandmasters, ຄອມພິວເຕີຈະເວົ້າໄດ້ຢ່າງຄ່ອງແຄວ ຄືກັບມະນຸດ. ເມື່ອເຮັດແນວນັ້ນຈະເຫັນໄດ້ວ່າທັກສະການສິນທະນາບໍ່ແມ່ນການພິສູດຄວາມສະຫຼາດ. ນັ້ນແມ່ນປັນຫາເພາະເຮົາບໍ່ສາມາດຮູ້ໄດ້ວ່າມີໃຜຄົດແນວໄດແດ່, ການສື່ສານເປັນສົງທີ່ເຮົາມີ.

1.2 ປະຫວັດຄວາມເປັນມາຂອງບັນຍາປະດິດ (History of AI)

ກ່ອນກຳເນີດ AI (ຄ.ສ. 1943-1955)

- ໂປຣແກຣມບັນຍາປະດິດເປັນໂປຣແກຣມທຳອິດໃນການຈຳລອງໜ່ວຍປະສາດດູວ (neuron) ສ້າງ ໂດຍ Warren McCulloch ແລະ Walter Pitts ໂດຍໃຊ້ຄວາມຮູ້ເລື່ອງໜ້າທີ່ຂອງສະໝອງໃນດ້ານກາຍພາບ, ຕັກສາດ. ຫິດສະດີການຄິດໄລ່ ແລະ ພາຍຫຼັງ Donald Hebb ໄດ້ສະເໜີກິດການຮຽນຮູ້ເພື່ອ ອະທິບາຍການຮຽນຮູ້ຂອງເຕືອຂ່າຍເສັ້ນປະສາດທຸກ
- ມາລົວິນ ມິນສົກີ (Marvin Minsky) ແລະ ດີນ ເອັດມອນດ (Dean Edmond) ນັກສຶກສາ ມະຫາວິທະຍາໄລ Princeton ໄດ້ຮ່ວມກັນສ້າງເຕືອຂ່າຍເສັ້ນປະສາດທຸກ (Neural Network) ໃຊ້ ຫຼອດສູນຍາກາດເຖິງ 3000 ຫຼອດ ຈຳລອງໜ່ວຍປະສາດ 40 ຫນ່ວຍ
- ອະລັນ ຫົວລົງ (Alan Turing) ໄດ້ສະເໜີວິທີການຫິດສອບວ່າໂປຣແກຣມສະໝາດ ຫຼື້ນໍ້າ, ວິທີນີ້ເອີ້ນວ່າ Turing Test, ເຕືອງຈັກຮຽນຮູ້, ການຮຽນຮູ້ແບບເສີມແຮງ ແລະ ອານກຳລືທີມດ້ານພັນຫຼຸກຳ (Genetic Algorithm). ໃນຊ່ວງນີ້ໄດ້ມີການສ້າງແນວຕິດອັນເປັນພື້ນຖານຂອງການສ້າງຜົນງານບັນຍາປະດິດ.

ການກຳເນີດ AI (ຄ.ສ. 1956)

- ບັນຍາປະດິດກຳເນີດຢ່າງເຕັມຕົວທີ່ມະຫາວິທະຍາໄລ Princeton ໂດຍ ຈອຫັນ ແມກາລາ໌ທີ (John McCarthy) ໄດ້ຊວນ ມາລົວິນ ມິນສົກີ (Marvin Minsky), ຄົດ ແຊນນອນ (Claude Shannon),, ນາທານຍົນ ໂລເຊດເຕີລ (Nathaniel Rochester) ແລະ ນັກວິຈະຈາກສະຖາບັນອື່ນລວມ 10 ຄົນ ໃຫ້ ຂ່ວຍກັນເຮັດວິໄຈເລື່ອງຫິດສະດີອັດຕະໂນມັດ (Automata Theory) ເຕືອຂ່າຍເສັ້ນປະສາດ ແລະ ສຶກສາເລື່ອງ "ຄວາມສະໝາດ: Intelligence"
- Allen Newell ແລະ Herbert Simon ໄດ້ພັດທະນາໂປຣແກຣມທາເຫດຜົນ ແລະ ພິສູດຫິດສະດີຕັກກະສາດ ຕີໂປຣແກຣມນັກຫິດສະດີຕັກສາດ (Logic Theorist)
- ຢູ້ຕັ້ງຊື່ໃຫ້ ກັບວິທະຍາສາດສາຂາໃໝ່ນີ້ຄື John McCarthy ກ່ອນທີ່ຈະໄດ້ຊື່ວ່າ AI: Artificial Intelligence ນັ້ນມີອີກຊື່ໜຶ່ງທີ່ໜ້າຈະເໝາະຄື Computational Rationality
- AI ກ່ຽວຂ້ອງກັບວິທະຍາສາດດ້ານຈົດຕະວິທະຍາ, ກາຍະວິພາກ, ຄະນິດສາດ, ແຕ່ເປັນວິທະຍາສາດ ຢ່ອຍຂອງວິທະຍາສາດຄອມພິວເຕີ ເນື້ອງຈາກເນັ້ນການເຮັດໃຫ້ເຕືອງຈັກສາມາດເຮັດວຽກທີ່ມະນຸດເຮັດໄດ້ ຫຼື ເຮັດໄດ້ດີກວ່າ.

ຄວາມຄາດຫວັງອັນຍິ່ງໃຫຍ່ທີ່ຕ້ອງການຈາກ AI (ຄ.ສ. 1952-1969)

- AI ໃນຊ່ວງທຳອິດນີ້ໃຫ້ວ່າປະສົບຄວາມສໍາເລັດ Newell ແລະ Simon ໄດ້ສ້າງອີກໂປຣແກຣມຕີໂປຣແກຣມແກ້ບັນຫາທີ່ໄປ (GPS: General Problem Solver) ຈຳລອງວິທີການແກ້ບັນຫາໂດຍທີ່ໄປຂອງມະນຸດ (Thinking Humanly) ໂດຍໄດ້ຫິດລອງກັບບັນຫາການຕໍ່ຄໍາ (Puzzle) ໃນຂອບເຂດຄວາມຍາກທີ່ກຳນົດ.
- McCarthy ໄດ້ສ້າງພາສາລະດັບສູງເພື່ອຊົງໂປຣແກຣມດ້ານບັນຍາປະດິດໂດຍສະເພາະ ນັ້ນຄືມີຄວາມສາມາດໃນການທາເຫດຜົນ ຈັດການກັບໂຈດບັນຫາທີ່ບໍ່ແມ່ນຕົວເລກ ພາສານີ້ຄື LISP

- McCullough และ Pitts สืบต่อวิจัยเดิมของสแตนเลสันและสาดทูม

ສິ່ງທີ່ເຮັດໄດ້ ແລະ ສິ່ງທີ່ເຮັດບໍ່ໄດ້ (ຄ.ສ. 1966-1973)

- ໂປຣແກຣມສ່ວນຫຼາຍບໍ່ມີຄວາມຮູ້ໃນຂົງເຂດຄວາມຮູ້ທີ່ຈະເອົາມາໃຊ້ໃນການແກ້ໄຂບັນຫາ
- ຄວາມສາມາດໃນການໂຕຕອບເຮັດໄດ້ຢາກ
- ແນວຕິດເລື່ອງເຕື່ອງຈັກກາຍພັນ
- Minsky ແລະ Papert ໄດ້ຊື້ໃຫ້ເຫັນເຖິງຂໍ້ຈຳກັດໃນການນຳໃຊ້ເຕື່ອຂ່າຍເສັ້ນປະສາດທຸກແບບໜຶ່ງຊັ້ນ, ເຊິ່ງສິ່ງຜົນໃຫ້ງານວິໄຈເລື່ອງນິ້ງງົບທາຍໄປນັ້ນ 10 ປີ, ເນື່ອງຈາກນັກວິໄຈເຊື່ອວ່າເຖິງທາງຕັ້ນແລ້ວ.

ລະບົບຖານຄວາມຮູ້ (ຄ.ສ. 1969-1979)

- ລະບົບຖານຄວາມຮູ້ເປັນການແທນຂໍ້ມູນໃຫ້ໂປຣແກຣມທີ່ເຮັດວຽກດ້ານ AI ສາມາດນຳໄປໃຊ້ຫາເຫັນ ຫຼືຫາຄຳຕອບໄດ້.
- ລະບົບສະໜັບສະໜູນການຕັດສິນໃຈເປັນການລວມຄວາມຮູ້ໃນຖານຄວາມຮູ້ກັບກົດເການເພື່ອຊ່ວຍໃນການຕັດສິນໃຈ.
- ລະບົບຜູ້ຂ່າຍວຊານໄດ້ຖືກຝັດທະນາຕໍ່ຈາກລະບົບສະໜັບສະໜູນການຕັດສິນໃຈ ໂດຍຮວບຮວມຄວາມຮູ້ໃນການແກ້ບັນຫາໜຶ່ງໆ ຈາກຜູ້ຂ່າຍວຊານໃນດ້ານນັ້ນໆ ແລະ ມີໂປຣແກຣມໃນການອ້າງເຫັດຜົນ. ຜູ້ໃຊ້ບ້ອນຂໍ້ມູນລັກສະນະຂອງບັນຫາເຂົາລະບົບ, ແລ້ວໂປຣແກຣມໃນການອ້າງເຫັດຜົນຈະຊອກຫາຄຳຕອບຫຼື ຄຳປຶກສາກັບຜູ້ໃຊ້.

ອຸດສາຫະກຳ AI (ຄ.ສ. 1980-ປັດຈຸບັນ)

ການໃຊ້ລະບົບຜູ້ຂ່າຍວຊານໃຫ້ຄຳປຶກສາໄດ້ເຂົ້າມາມີບົດບາດໃນວົງການອຸດສາຫະກຳ, ເຊັ່ນ: ລະບົບ R1 ຖືກໃຊ້ໃນບໍລິສັດ DEC ໃນການຊ່ວຍຫາການສັງເຊີ້ລະບົບຄອມພິວເຕີໃໝ່ທີ່ເໝາະສົມ, ເຊິ່ງເຮັດໃຫ້ບໍລິສັດປະຫຍັດໄດ້ເຖິງປີລະ 40 ລ້ານດອນລາ. ໃນບໍລິສັດ DEC ໃຊ້ໂປຣແກຣມລະບົບຜູ້ຂ່າຍວຊານເຖິງ 40 ລະບົບ. ບໍລິສັດອື່ນອີກໝາຍບໍລິສັດຕ່າງກຳໃຫ້ຄວາມສົນໃຈ ແລະ ຍອມລົງທຶນມະຫາສານສ້າງໂປຣແກຣມປັນຍາປະດິດ, ແຕ່ບໍ່ປະລືບຄວາມສຳເລັດຕ່າງໆທີ່ຄາດໄວ້.

ເຕືອຂ່າຍເສັ້ນປະສາດທຸກ -Neural Network (ຄ.ສ. 1986-ປັດຈຸບັນ)

- ຕັ້ງແຕ່ປີ 1970 ເຕືອຂ່າຍເສັ້ນປະສາດທຸກໄດ້ຮັບຄວາມສົນໃຈນັ້ນອຍໝາຍ ເນື່ອງຈາກນັກວິໄຈເຊື່ອວ່າບໍ່ສາມາດສ້າງໂປຣແກຣມທີ່ແກ້ບັນຫາໄດ້ແທ້, ແຕ່ເນື່ອນນັກຝິຊີກຊີ Hopfield ໄດ້ໃຊ້ວິທີການທາງສະຖິຕິກົມລະສາດວິເຄາະຄວາມຕ້ອງການໜ່ວຍຄວາມຈຳ ແລະ ອຸນສົມບັດທີ່ເໝາະສົມທີ່ສຸດຂອງເຕືອຂ່າຍໄດ້ຍິ່ງແຕ່ລະໜ່ວຍໃນເຕືອຂ່າຍເປັນອາຕອມ, ເຮັດໃຫ້ງານວິໄຈເຕືອຂ່າຍເສັ້ນປະສາດທຸກໄດ້ກັບມາທີ່ກົມ
- ການຮຽນຮູ້ແບບແພ່ກະຈາຍກັບຄືນ (Backpropagation Learning) ໄດ້ຖືກສະເໜີໂດຍ Rumelhart, Hinton ແລະ Williams ສາມາດແກ້ບັນຫາໄດ້ທີ່ວ່າໄປໝາຍຂຶ້ນ ແລະ ເຮັດໃຫ້ນັກວິໄຈກັບມາໃຫ້ຄວາມສົນໃຈເຕືອຂ່າຍເສັ້ນປະສາດທຸກອີກຄົງ.

ວິທະຍາສາດ AI (ຄ.ສ. 1987-ປັດຈຸບັນ)

ບັນຍາປະດິດໄດ້ຮັບການຍອມຮັບເປັນວິທະຍາສາດສາຂາໜຶ່ງໃນປີ 1987 ເນື່ອງຈາກທີ່ຜ່ານມາມີການຄົດຄົ້ນວິທີການ, ທິດສະດີທີ່ເຮັດໃຫ້ການສ້າງເຄື່ອງຈຳກັບມີຄວາມສະໜາດ, ແກ້ວມໜາໄດ້ຈິງ. ເຊັ່ນ: ຫຼຸ່ມຍິນ, ຄອມພິວເຕີວິຊັນ (Computer Vision), ການແຫນຄວາມຮູ້, ການຮູ້ຈີ່ຈໍາສົງ, ການຄົ້ນຫາຄວາມໝາຍທີ່ມີຢູ່ໃນຂໍ້ມູນຈໍານວນໝາຍ (Data Dining) ເປັນຕົ້ນ.

ຕົວແທນບັນຍາປະດິດ (ຄ.ສ. 1995-ປັດຈຸບັນ)

ໃນການແກ້ວມໜາຂອງໂປຣແກຣມບັນຍາປະດິດນີ້ ຖ້າເປັນບັນຫາຂໍ້ບ້ອນ ແລະ ບັນຫານີ້ມີການຄອຍຖ້າຕິດຕໍ່ ຫຼື ຕິດຕາມເບິ່ງການປ່ຽນແປງຕ່າງໆຂອງສິ່ງແວດລ້ອມ ມັກຈະແບ່ງວູກອອກເປັນວູກຍ່ອຍ, ແລ້ວມີຕົວແທນບັນຍາ (Intelligent Agent) ເຮັດວຽກໃນສ່ວນຍ່ອຍ. ຕົວຢ່າງທີ່ເຫັນໄດ້ຈັງທີ່ສຸດຄືຕົວແທນບັນຍາເທິງອິນເຕີເນັດ, ເຊິ່ງເຮົາມັກຈະເຮັ້ນວ່າ bot (ບອດ) ເຊັ່ນ: ບອດຂອງໂປຣແກຣມຄົ້ນຫາຂໍ້ມູນ (Search Engine).

1.3 ການປະຍຸກໃຊ້ AI ໃນປັດຈຸບັນ

ປັດຈຸບັນງານວິໄຈທາງດ້ານບັນຍາປະດິດ ໄດ້ມີການນຳໄປປະຍຸກໃຊ້ໃນການແກ້ໄຂບັນຫາຕ່າງໆຢ່າງກວ້າງຂວາງ ໃນໝາຍໆສາຂາ, ແຕ່ວູກສ່ວນໃຫຍ່ຈະເນັ້ນໜັກໄປໃນຮູບແບບທີ່ໃຊ້ເຫດຜົນເປັນຫຼັກ. ເນື່ອງຈາກສາຂາທີ່ມີການນຳບັນຍາປະດິດໄປປະຍຸກໃຊ້ແກ້ບັນຫາ ບໍ່ຈໍາເປັນຕ້ອງອາໄສອາລີມ ຫຼື ຄວາມຮູ້ສີກຂອງມະນຸດ. ຕໍ່ໄປນີ້ຈະກ່າວເຖິງສາຂາເຊິ່ງເປັນທີ່ຮູ້ຈັກກັນຢ່າງແພ່ງໝາຍໄດ້ແກ່:

- ການປະມວນຜົນພາສາທຳມະຊາດ (Natural Language Processing)

ເປັນການສຶກສາທີ່ເນັ້ນໃຫ້ຄອມພິວເຕີສາມາດສື່ສານກັບຜູ້ໃຊ້ໄດ້ໂດຍໃຊ້ພາສາທຳມະຊາດ ການວິໄຈສາຂານີ້ຕ້ອງອາໄສຄວາມຮູ້ທາງດ້ານພາສາສາດ ເພື່ອສຶກສາເຖິງວິທີການປະມວນຜົນລວມທັງການຮັບຮູ້, ການເຂົ້າໃຈ ແລະ ການນຳໃຊ້ພາສາທຳມະຊາດ ເພື່ອໃຫ້ຄອມພິວເຕີສາມາດເຂົ້າໃຈພາສາມະນຸດໄດ້. ຕົວຢ່າງງານວິໄຈດ້ານນີ້ໄດ້ແກ່: ເຄື່ອງແປພາສາ, ລະບົບຄົ້ນຄົ້ນໂດຍໃຊ້ພາສາທຳມະຊາດ, ລະບົບສອບຖາມທາງໂທລະສັບອັດຕະໂນມັດ.

- ວິທະຍາການຫຼຸ່ມຍິນ (Robotics)

ໂດຍທີ່ວ່າປ່ຽນຍິນໝາຍເຖິງເຄື່ອງຈຳທີ່ສາມາດຮັດວຽກໃນທາງກາຍພາບຕ່າງໆໄດ້, ເຊິ່ງຫຼຸ່ນຍືນເຫຼົ່ານີ້ຈະປະກອບດ້ວຍ Effectors ເຊັ່ນ: ຂາວົງລົ້, ຂັ້ຕໍ, ຕົວຍິດຈະບ ເຊິ່ງສ່ວນທີ່ເຮັດໜ້າທີ່ສົງແຮງອອກໄປທາສະພາບແວດລ້ອມ ແລະ Sensors ທີ່ເປັນຕົວຮັບຮູ້ສະພາບແວດລ້ອມ. ການວິໄຈໃນສາຂານີ້ຈະເປັນຕ້ອງອາໄສຄວາມຮູ້ທາງກົນຈັກ ເພື່ອຮັດໃຫ້ຫຼຸ່ມຍິນສາມາດເຄື່ອນໄຫວໄດ້ຕາມທີ່ໂປຣແກຣມຄວບຄຸມສັງການ. ປັດຈຸບັນຫຼຸ່ມຍິນອາດຈະແບ່ງໄດ້ເປັນ 3 ກຸ່ມຫຼັກໆ, ກຸ່ມທຳອິດໄດ້ແກ່: ແຂນກົນ ຫຼື Manipulators ເຊິ່ງສ່ວນໃຫຍ່ມັກຈະໃຊ້ໃນວົງການອຸດສາຫະກຳ, ເຊັ່ນ: ວູກຍົກເຄື່ອງ, ວູກປະກອບຊື້ນສ່ວນຕ່າງໆ ຫຼື ລວມທັງໃນວົງການແພດກົມີແຂນກົນທີ່ຊ່ວຍໃນການຜ່າຕັດ. ກຸ່ມທີ່ສອງເຮັ້ນວ່າ ຫຼຸ່ມຍິນເຄື່ອນທີ່ ຫຼື Mobile Robot ເຊິ່ງສາມາດເຄື່ອນໄຫວໃຫ້ເຂົ້າກັບສິ່ງແວດລ້ອມໄດ້, ຕົວຢ່າງເຊັ່ນ: ຫຼຸ່ມຍິນທີ່

ສາມາດເຄື່ອນທີ່ໄດ້ເຫັງພື້ນຜົວ, ໄຕນຳໍ້ ຫຼື ໃນອາກາດ ແລະ ກຸ່ມສຸດທ້າຍແມ່ນຫຼຸ່ມຍິນແບບປະສົມ ຫຼື hybrid ທີ່ລວມເອົາຄວາມສາມາດຂອງຫຼຸ່ມຍິນສອງກຸ່ມທຳອິດເຂົ້າກັນ, ໂດຍລວມທັງຫຼຸ່ມຍິນທີ່ເຮັ້ນວ່າ Humanoid ເຊິ່ງເປັນຫຼຸ່ມຍິນທີ່ອອກແບບໃຫ້ມີຮູບຮ່າງຄ້າຍຄືມະນຸດ ຕົວຢ່າງທີ່ມີຊື່ສົງຂອງຫຼຸ່ມຍິນແບບນີ້ ໄດ້ແກ່ ASIMO ຂອງບໍລິສັດຮອນດ໏າ.

- ການພິສູດທຶນສະດີ (Theorem Proving)

ເປັນການພັດທະນາໂປຣແກຣມຄອມພິວເຕີເພື່ອໃຊ້ພິສູດທິດສະດີໄດ້ຍອາໄສກົດເກມທາງຕັກສາດ (Predicate Logic) ເຊິ່ງສາມາດນຳໄປໃຊ້ການພິສູດທິດສະດີບົດທາງຄະນິດສາດ ຫຼື ນຳໄປໃຊ້ໃນການກວດສອບການອອກແບບວິຈອນ.

- ລະບົບຜູ້ຊ່າວຊານການ (Expert Systems)

เป็นงานวิจัยที่เน้นส้าง拉斯บัญช่วงงานเพื่อให้คำปรึกษา และ ดำเนินการทดสอบกู่วัวบันชาต่างๆ, โดยจะใช้แบบสอบถามความรู้ และ ข้อมูลจากประสบการณ์ของผู้เชี่ยวชาญ และ สามารถนำความรู้นั้นมาวิเคราะห์เพื่อหาดำเนินการทดสอบของบันชา. งานวิจัยฯ มีจุดประสงค์เพื่อทดสอบความรู้พื้นฐานหลายอย่าง เช่น: ภาระแบบสอบถามความรู้, ภาระคิดให้เหตุผล และ ภาระที่ต้องรู้ด้วยตัวเอง. ตัวอย่าง งานวิจัยฯ ได้รับการสนับสนุนจากหน่วยงานต่างๆ 例如: สถาบันวิทยาศาสตร์แห่งประเทศไทย (MYCIN) ที่ให้ข้อมูลทางพัฒนาการแก่เด็ก สถาบันวิทยาศาสตร์แห่งประเทศไทย (DENDRAL) ที่สามารถวิเคราะห์สังเคราะห์光譜ограмม์ (Spectrogram) ของมวนต่างๆ.

- ការងារខ្សោយប្រព័ន្ធអំពីការអ៊ិចធីសម្រាប់ប្រព័ន្ធ (Automatic Programming)

เป็นภารกิจของผู้เขียนโปรแกรมให้คำสั่งโปรแกรมได้อ่านและเข้าใจภาษาที่เขียนมา โดยสามารถแปลงภาษาที่เขียนมาเป็นภาษาที่คอมพิวเตอร์เข้าใจได้ เช่นภาษา C หรือภาษา Pascal เป็นต้น ซึ่งภาษาที่เขียนมาจะถูกแปลงเป็นภาษาที่คอมพิวเตอร์เข้าใจได้โดยอัตโนมัติ ทำให้เราสามารถเขียนโปรแกรมโดยไม่ต้องรู้รายละเอียดการทำงานของเครื่องคอมพิวเตอร์มากนัก แต่ต้องรู้รายละเอียดของภาษาที่ใช้เขียน เช่นโครงสร้างข้อมูล คำสั่งควบคุม ฯลฯ ที่สำคัญคือ ผู้เขียนโปรแกรมต้องเขียนคำสั่งในภาษาที่คอมพิวเตอร์เข้าใจได้ ไม่ใช่ภาษาธรรมชาติ เช่นภาษาไทย ภาษาอังกฤษ ฯลฯ แต่เป็นภาษาที่เรียกว่าภาษาเครื่อง หรือภาษาโปรแกรม ที่มีรูปแบบและ構造ที่เฉพาะเจาะจง ไม่สามารถอ่านและเข้าใจได้โดยตรง แต่ต้องผ่านการแปลงจากภาษาธรรมชาติที่เขียนมาเป็นภาษาที่คอมพิวเตอร์เข้าใจได้

- ບັນຫາການຈັດຕາຕະລາງ (Scheduling Problems)

เป็นภารນ์มาอิ่มย่าปะดีดໄบใช้ในການແກ້ບັນຫາການກຳນົດຕາຕະລາງເວລາ ຫຼື ການເລືອກ Combination ໃຫ້ເໝາະສົມທີ່ສຸດ, ເຊັ່ນ: ການຈັດຕາລາງການເດີນຫາງຂອງພະນັກງານຂາຍເຄື່ອງໂດຍ ເສຍຄ່າໃຊ້ຈ່າຍນ້ອຍທີ່ສຸດ, ການຈັດຕາລາງການຂຶ້ນລົງຂອງຍິນໃຫ້ເກີດປະໂຫຍດສູງສຸດ, ການຈັດຕາລາງການຮຽນການສອນໃຫ້ເກີດປະສິດທິພາບສູງສຸດ ເປັນຕົ້ນ.

- ບັນຫາດ້ານປະສາດສຳຜັດ (Perception Problems)

ເປັນການປະຢຸກໃຊ້ເພື່ອໄປແກ້ບັນຫາການຮັບຮູ້ຕ່າງໆ ເຊັ່ນ: ການເບິ່ງ-ເຫັນ, ການສຳຜັດ, ການຟັງຕົວຢ່າງເຊັ່ນ: ການສ້າງຫຸ່ມຍືນທີ່ສາມາດຍ່າງຫລືບຫລືກສິ່ງກິດຂວາງໄດ້.

❖ ສັ່ງທີ່ປັນຍາປະດິດຍ້າຮັດບໍ່ທັນໄດ້ເຫຼືອ

- ເຊິ້ງໃຈພາສາກຳມະນູາດໄດ້ຢ່າງດີ ເຊັ່ນ: ອ່ານ ແລະ ເຊິ້ງໃຈທີ່ວ່າງໆໃນໜ້າສີພົມ

- ການທ່ອງໄລກອິນເຕີເມັດ (Surf the web)
- ຕົກວາມໝາຍຂອງຮູບພາບຕາມໃຈ
- ຮຽນພາສາທຳມະຊາດ
- ສ້າງແຜນໃນຂອບເຂດເລາຈິງໄດ້ຢ່າງຄ່ອງແຄວ
- ສະແດງໃຫ້ເຫັນລັກສະນະ ແລະ ຄວາມສະໜາດທີ່ແທ້ຈິງ

1.4 ປະເພດຂອງບັນຍາປະດິດ (AI Category)

ໃນຕາຕະລາງລຸ່ມນີ້ແມ່ນຕົວຢ່າງຂອງບັນຍາປະດິດ, ເຊິ່ງສ່ວນຫຼາຍແມ່ນບັນຍາປະດິດຢູ່ກາໃໝ່.

ປະເພດ	ຄໍາອະທິບາຍ
Affective Computing	ການສຶກສາແລະການພັດທະນາຂອງລະບົບ ແລະ ອຸປະກອນ ທີ່ສາມາດຮັບຮູ້, ຕົກວາມໝາຍ, ດຳເນີນການ, ແລະ ການຈໍາລອງ ຄວາມເປັນມະນຸດ.
Artificial Immune Systems	ລະບົການຮຽນຮູ້ແບບຮັດສະລິຍະທີ່ອີງໃສ່ຫຼັກການ ແລະ ຂະບວນການເຊິ່ງມີຢູ່ໃນລະບົບພູມຕ້ານທານຂອງສັດລັງລູກຄ້ວຍນິມ
Chatterbot	ປະເພດຂອງຕົວແທນການສົນທະນາ ຫຼື ໂປຣແກຣມຄອມພິວເຕີທີ່ອກແບບມາເພື່ອກະຕຸນການສົນທະນາຮັດສະລິຍະກັບຜູ້ໃຊ້ຄົນໜຶ່ງຫຼືຫາຍຄົນຜ່ານຂໍ້ຄວາມຫຼືສູງ
Cognitive Architecture	ທີ່ດສະດີກ່ຽວກັບ ໂຄງສ້າງຂອງຈິດໃຈມະນຸດ. ໜຶ່ງໃນເປົ້າໝາຍຫຼັກຄືການໂຮມແນວຄົດຈາກຈິດຕະສາດ, ຄວາມຮູ້ສຶກ, ຄວາມເຂົ້າໃຈ ໃສ່ໃນແບບຈໍາລອງຄອມພິວເຕີທີ່ຄວບຄຸມ
Computer Vision	ສາຂາວິທະຍາສາດແບບກວ້າງທີ່ກ່ຽວຂ້ອງກັບວິທີການທີ່ຄອມພິວເຕີສາມາດຮັບຄວາມເຂົ້າໃຈລະດັບສູງຈາກຮູບພາບ ແລະ ວິດໂອ.
Evolutionary Computing	ການໃຊ້ຂັ້ນຕອນວິທີທີ່ມີວັດທະນາການຕາມຫຼັກການຂອງ ດາວິນຈຸນ (Darwinian)ເຊິ່ງເປັນທີ່ມາຂອງຊື້ນີ້. ຂັ້ນຕອນວິທີເຫຼົ່ານີ້ຢູ່ໃນຕະກຸນຂອງຕົວແກ້ບັນຫາແບບການທີ່ດລອງແລະຂໍຜິດພາດ (trial-and-error) ແລະ ໄຊວິທີ ເມຕ້າຮິວຮິດສະຕິກ (metaheuristics) ຫຼື stochastic global ເພື່ອກຳນົດການແກ້ໄຂແບບຕ່າງໆ
Gaming AI	AI ໃຊ້ໃນເກມເພື່ອສ້າງພິດຕິກຳສະໜາດ, ໂດຍສ່ວນຫຼາຍຢູ່ໃນຕົວທີ່ບໍ່

	ແມ່ນຜູ້ຫີ່ນ (non-player characters :NPCs) ເຊິ່ງມໍາຈະຈໍາລອງຄວາມສະຫຼາດຄື່ນນຸດ.
Human-Computer-Interface (HCI)	HCI ຄົ້ນຄ້ວາການອອກແບບ ແລະ ການນຳໃຊ້ເຕັກໂນໂລຊີຄອມພິວເຕີ, ເນັ້ນໃສ່ການເຊື່ອມຕໍ່ລະຫວ່າງຄົນ (ຜູ້ໃຊ້) ແລະ ຄອມພິວເຕີ.
Intelligent Soft Assistant or Intelligent Personal Assistant (IPA)	ຕົວແທນຊອບແວທີສາມາດປະຕິບັດວຽກຫຼຶບລິການສໍາລັບບຸກຄົນໄດ້ຫຍື່ງ. ຫ້າວຽກ ຫີ້ ການປໍລິການ ເຖິງນີ້ຕາມປົກກະຕິແມ່ນອີງໃສ່ການປ້ອນຂໍ້ມູນຂອງຜູ້ໃຊ້, ການຮັບຮູ້ສະຖານທີ່ ແລະ ຄວາມສາມາດໃນການເຂົ້າເຖິງຂໍ້ມູນຈາກແຫຼ່ງຂໍ້ມູນອອນໄລນ໌. ຕົວຢ່າງຂອງຕົວແທນດັ່ງກ່າວແມ່ນ Apple's Siri, Amazon's Alexa, Amazon's Evi, Google's Home, Microsoft Cortana, ແຫຼ່ງເປີດ Lucida, Braina (ແອບພິເຕເຊັ້ນທີ່ພັດທະນາໂດຍ Brainasoft ສໍາລັບ Microsoft Windows), S Voice ຂອງ Samsung ແລະ LG G3 Voice Mate.
Knowledge Engineering	ໜາຍເຖິງຫັ້ງໝົດຂອງ ດ້ວຍວິຊາການ, ວິທະຍາສາດແລະສັງຄົມທີ່ກ່ຽວຂ້ອງໃນການສ້າງ, ການຮັກສາແລະ ນຳໃຊ້ລະບົບຄວາມຮູ້.
Knowledge Representation	ເຊິ່ງເປັນຕົວແທນຂອງຂໍ້ມູນກ່ຽວກັບ ໂລກໃນຮູບແບບທີ່ລະບົບຄອມພິວເຕີສາມາດນຳໃຊ້ເພື່ອແກ້ໄຂວຽກງານທີ່ສັບສົນເຊັ່ນ: ການວິນິດໄສສະພາບທາງການແພດຫຼືມີການສົນທະນາໃນພາສາທຳມະຊາດ.
Logic Programming	ປະເພດຂອງການຂຽນໂປຣແກຣມສ່ວນຫຼາຍແມ່ນອີງໃສ່ຕາມຫຼັກຕັກກະສາດ. ໂປຣແກຣມໄດ້ງໍທີ່ຂຽນໃນພາສາໂປແກ້ມແບບບັດກະສາດເປັນຊຸດຂອງປະໂຫຍກໃນແບບຕັກກະສາດ, ສະແດງຂໍ້ເທັດຈິງແລະກົດກ່ຽວກັບບາງຂອບເຂດບັນຫາ. ຕະກູນຫຼັກງານຂອງພາສາໂປແກ້ມແບບບັດກະສາດມີ ພາສາ PROLOG, Answer Set Programming (ASP) ແລະ Datalog.
Machine Learning	ML ໃນເນື້ອໃນຂອງ AI ໃຫ້ຄອມພິວເຕີສາມາດຮຽນຮູ້ໄດ້ເອງໂດຍບໍ່ມີການຕັ້ງໂປຣແກຣມໄວ້ຢ່າງຊັດ. ການຮຽນຮູ້ແບບຕົ້ນແລະແບບເລີກເປັນສາຂາຍ່ອຍທີ່ສຳຄັນ.
Multi-Agent System	MAS ຄືລະບົບທີ່ເຮັດໃຫ້ Agent ຫ້າຍໆເຊື່ອງເຮັດວຽກຮ່ວມກັນ, ເຊິ່ງ Agent ເຖິງນີ້ແຕ່ລະໄຕສາມາດເຮັດວຽກໄດ້ຢ່າງອິດສະຫຼະ ແລະທີ່ສຳຄັນແຕ່ລະໄຕມາດຕັດໃຈໄດ້ດ້ວຍຕົວເອງ
Robotics	Robotics ເປັນສາຂາທີ່ໃຫຍ່ທ່າງດ້ານວິສະວະກຳ ແລະ ວິທະຍາສາດທີ່

1.5 ບັນຍາປະດິດ ແລະ ບີກາຕົ້າ (AI and Big data)

ເຮົາອາດຈະເຄີຍໄດ້ຢືນຄໍາວ່າ *Big Data*, ແຕ່ວ່າຄົນສ່ວນຫຼາຍອາດຈະບໍ່ຮູ້ວ່າມັນແມ່ນຫຍັງ ແລະ ມີຜົນກະທິບໍຍັງຕໍ່ກັບສັງຄົມຍຸກໃໝ່. *Big Data* ມີຫຼາຍນິຍາມ, ຄືກັນກັບປັນຍາປະດິດທີ່ມີຫຼາຍນິຍາມຄືກັນ. ນິຍາມຕໍ່ໄປນີ້ເປັນນິຍາມແບບທີ່ວ່າງໄປ: ຊຸດຂໍ້ມູນທີ່ປະກອບມີ 3ອົງປະກອບຄື: ຂໍ້ມູນຂະໜາດໃຫຍ່, ຄວາມໄວ ແລະ ຄວາມຫຼາກຫຼາຍ (*a data collection characterized by huge volumes, rapid velocity, and great variety*).

ປະກອບມີໜ່ວຍປມວນຜົນຫຼາຍພັນ ໂຕທີ່ເຮັດວຽກຂະໜານກັນເພື່ອເພີ່ມຄວາມໄວຂອງຂະບວນ, ການຫຼຸດຂໍ້ມູນທີ່ຖືກເອີ້ນວ່າ MapReduce. ຄອມພິວເຕີ IBM's Watson ເປັນລະບົບຕົວຢ່າງທີ່ສຳຄັນ, ມີການໃຊ້ລະບົບຜູ້ຊົງວ່າມານທາງການແພດໄດຍ໌ໃຊ້ Rule-Based Engine ແລະ ປະມວນຜົນຂໍ້ມູນທາງການແພດຫຼາຍພັນ Record ເກືອບວ່າຮອດລ້ານ, ຜົນທີ່ໄດ້ຄືລະບົບຄອມພິວເຕີທີ່ຊ່ວຍແພດບົ່ງມະຕິພະຍາດ ແລະ ພະຍາດກົງວຂ້ອງທີ່ບໍ່ສະແດງອາການ ຫຼື ມີຄວາມກົງວຂ້ອງກັບພະຍາດທີ່ຮູ້.

บิ๊ดที 2 ภาระแก้ปัญหาด้วยการนิวเคลียร์ (Problem Solving using Search)

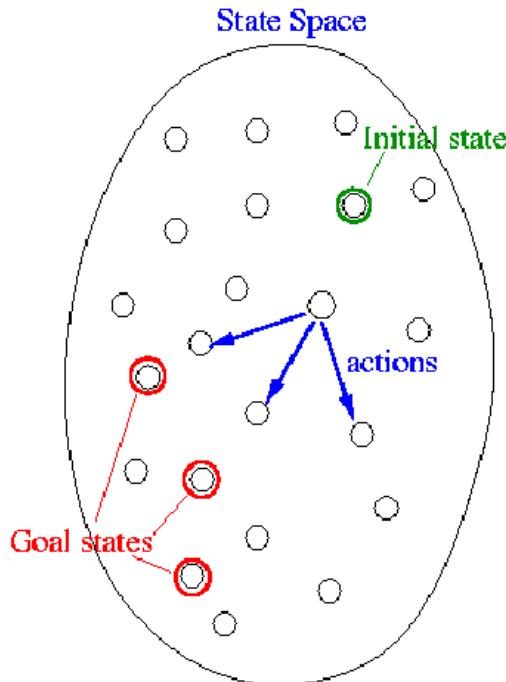
2.1. ຂອບເຂດຂອງປັນຫາ (State Space)

เมื่อปี 1976 เมวิล และ ไนมอน (Newell and Simon, 1976) ได้สังเขปแบบวิถีกู่วอกับภาระคิดว่า ภาระคิดที่เป็นอันดับสั้นจะช่วยให้ความสามารถในการแก้ไขข้อความที่ไม่ถูกต้องลดลง

- ການໃຊ້ຮູບແບບຂອງສັນຍາລັກ (Symbolic Pattern) ແທນຄ່າອີງປະກອບທີ່ສໍາຄັນຂອງປັນຫາໃນຂອບເຂດຂອງຄວາມຮູ້ນັ້ນ.
 - ການປະມວນຜົນຮູບແບບຂອງສັນຍະລັກເຖິງນີ້ເພື່ອຫາຄໍາຕອບທີ່ເປັນໄປໄດ້ (Possible Solution) ຫ້າງໝົດອອກມາ.
 - ຄົ້ນຫາເພື່ອເລືອກຄໍາຕອບຈາກຄໍາຕອບທີ່ເປັນໄປໄດ້ຫ້າໝົດທີ່ຖືກສ້າງຂຶ້ນມາໃນຂັ້ນຕອນກ່ອນໜ້ານັ້ນ.

ຂໍສະຫຼຸບຂອງເນວີວ ແລະ ໄຊມອນ ເຮັດໃຫ້ເກີດການແກ້ປັນຫາແບບ ຂອບເຂດສະຖານະ (State Space) ເຊິ່ງເປັນວິທີການແກ້ປັນຫາແບບໜຶ່ງທີ່ກຳນົດວ່າປັນຫາມີຈຸດເລີ່ມຕົ້ນ ແລະ ເປົ້າໝາຍ ຂອງການແກ້ປັນຫາ, ເຊິ່ງເອີ້ນວ່າສະຖານະຈຸດເລີ່ມຕົ້ນ ຫຼື ສະຖານະເລີ່ມຕົ້ນ (Initial State) ເປັນສະຖານະທີ່ກຳນົດຈຸດເລີ່ມຕົ້ນຂອງການແກ້ປັນຫາ ຫຼື ໂຈດຂອງປັນຫາ, ສໍາລັບເປົ້າໝາຍ ຫຼື ສະຖານະເປົ້າໝາຍ (Gold State) ຄືສິ່ງທີ່ເຮົາຄາດຫວັງຈະໄດ້ຈາກການແກ້ປັນຫານີ້ຄໍາຕອບ. ສະຖານະທັງສອງນີ້ຈະຢູ່ຫ່າງຈາກກັນແລະ ສິ່ງທີ່ເຮົາຈະຫາທາງໄປຈາກສະຖານະເລີ່ມຕົ້ນມາສະຖານະເປົ້າໝາຍໄດ້ຈະຕັ້ງຜ່ານສະຖານະຕ່າງໆທີ່ຢູ່ລະຫວ່າງກາງຈຳນວນໝາຍ ແລະ ສະຖານະເຫຼົ້ານີ້ສາມາດທຽບໄດ້ກັບຄໍາຕອບຍ່ອຍໆທີ່ເຮົາໄດ້ຮັບຈາກການແກ້ປັນຫາກ່ອນທີ່ຈະໄດ້ຄໍາຕອບສຸດຫ້າຍ, ສະຖານະເຫຼົ້ານີ້ລວມກັນເອີ້ນວ່າ ພື້ນທີ່ປັນຫາ (Problem Space). ໃນພື້ນທີ່ປັນຫາຈະມີບາງສະຖານະເຫຼົ້ານັ້ນທີ່ສາມາດເປັນຫາງຜ່ານໄປສູ່ເປົ້າໝາຍ ຫຼື ຄໍາຕອບສຸດຫ້າຍໄດ້, ແຕ່ບາງສະຖານະຈະບໍ່ສາມາດພາໄປສູ່ເປົ້າໝາຍ ຫຼື ຄໍາຕອບສຸດຫ້າຍໄດ້, ຄວາມຫ່າງຂອງສະຖານະເລີ່ມຕົ້ນ ແລະ ສະຖານະເປົ້າໝາຍນີ້ກຳນົດໄດຍ່ຈຳນວນສະຖານະທີ່ຢູ່ລະຫວ່າງກາງ ຖ້າສະຖານະເລີ່ມຕົ້ນ ແລະ ສະຖານະເປົ້າໝາຍຫ່າງກັນໝາຍ ໝາຍຄວາມວ່າຂະໜາດຂອງພື້ນທີ່ປັນຫານີ້ກໍຈະໃຫຍ່ ຖ້າຄວາມຫ່າງນີ້ມອຍຂະໜາດພື້ນທີ່ປັນຫາກໍຈະນີ້ຍ. ເມື່ອເລີ່ມແກ້ປັນຫາ ການເຮັດວຽກຂອງລະບົບຈະຢູ່ທີ່ສະຖານະເລີ່ມຕົ້ນ ຈາກນັ້ນຈະຕັ້ງພະຍາຍາມຫາກາງໄປຈາກສະຖານະ

ເລີ່ມຕົ້ນນີ້ຜ່ານສະຖານະຕ່າງໆທີ່ຢູ່ລະຫວ່າງກາງ ເພື່ອໄປສູ່ສະຖານະເວົ້າໝາຍໃຫ້ໄດ້ ເມື່ອສາມາດໄປໜາສະຖານະເວົ້າໝາຍໄດ້ ການແກ້ບັນຫາກໍຈະສິນນສຸດຄືດັ່ງສະແດງໃນຮູບທີ 2.1



ຮູບທີ 2.1 State Space

ປົງທູບການແກ້ບັນຫາຂອງພື້ນທີ່ບັນຫາເປັນຮູບປະທຳກໍແມ່ນການຫາທາງໄປເຂົາວົງກົດສະຖານະເລີ່ມຕົ້ນຄືທາງເຂົ້າ ແລະ ສະຖານະເວົ້າໝາຍຄືທາງອອກ, ສໍາລັບເສັ້ນທາງຕ່າງໆໃນເຂົາວົງກົດແມ່ນສະຖານະຕ່າງໆ, ໃນກໍລະນີນີ້ເຮົາຈະເຫັນໄດ້ວ່າມີເສັ້ນທາງບາງເສັ້ນທາງເທົ່ານັ້ນທີ່ພາໄປສູ່ທາງອອກ ແລະ ເສັ້ນທາງສ່ວນໝາຍຈະເປັນເສັ້ນທາງທີ່ພາໃຫ້ຫຼົງ, ເຊັ່ນດູວກັນກັບພື້ນທີ່ບັນຫາ, ເສັ້ນທາງພື້ນທີ່ບັນຫາຈະມີໝາຍ ແຕ່ມີໝາງບາງເສັ້ນທາງເທົ່ານັ້ນທີ່ພາໄປສູ່ຄໍາຕອບໄດ້.

ໃນການແກ້ບັນຫາແບບຂອບເຂດສະຖານະ (State Space) ເປັນການແກ່ນບັນຫາທັງໝົດອອກມາໃນຮູບແບບຂອງ ກຣາບຈະມີໂນດທີ່ສະແດງສະຖານະຕ່າງໆ, ການຫາຄໍາຕອບຈະອາໄສຫຼັກການຂອງເຄື່ອງຈໍາກັດສະຖານະ (Finite State Machine) ໃນການດໍາເນີນງານ

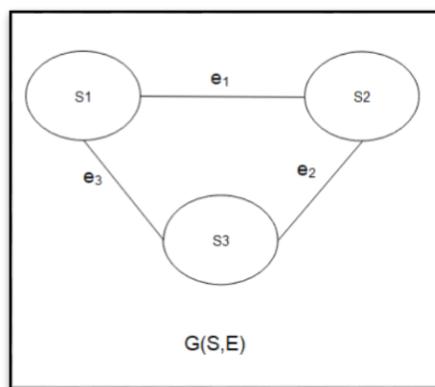
2.1.1 ການບັນຫາ (Graph)

ຄືໂຄງສ້າງຂໍ້ມູນແບບບໍ່ເປັນເສັ້ນຊື່ (Non-Linear) ທີ່ປະກອບມີ ໂນດ (Node) ຫຼື ສະຖານະ ແລະ ເສັ້ນເຊື່ອມ (Edge ຫຼື Link), ເຊິ່ງໂນດໝາຍເຖິງສິ່ງທີ່ສາມາດກຳນົດໄດ້ ແລະ ເກັບຂໍ້ມູນໄດ້, ສໍາລັບເສັ້ນເຊື່ອມແມ່ນເສັ້ນທີ່ເຊື່ອມກັນລະວ່າງໂນດ. ການກຳນົດທາງໄປຈາກໂນດໜຶ່ງໄປໜາອີກໂນດໜຶ່ງໄດ້ຍິ່ງທີ່ເສັ້ນທາງຈະບໍ່ຊັ້ງກັນ ໃນໂຄງສ້າງກຣາບຈະເອີ້ນວ່າ ເສັ້ນທາງ (Path). ກຣາບໜຶ່ງຈະມີໂນດພິເສດເອີ້ນວ່າ ອາກ (Root), ເຊິ່ງໜາຍເຖິງໂນດທີ່ມີເສັ້ນທາງຈາກໂຕມັນໄປໜາໂນດອື່ນຢູ່ເທິງກຣາບໄດ້ທຸກໂນດ, ຖ້າຫາກວ່າເສັ້ນທາງຈາກໂນດທາງໄປໜາທຸກໂນດໃນກຣາບມີໝາງເສັ້ນທາງດູວ ຈະເອີ້ນກຣາບນັ້ນ

ວ່າ ຕົ້ນໄມ້ (Tree) ແຕ່ຖາມີເສັ້ນທາງຈາກຮາກໄປຢ້າງໂນດອື່ນຫຼາຍກ່ວ່າໜຶ່ງເສັ້ນທາງຈະເອີ້ນໄຄງສ້າງນີ້ ວ່າ ກຣາບ.

ໂດຍທົ່ວໄປ ໃນໄຄງສ້າງຕົ້ນໄມ້ແລະກຣາບ ຈະຊູນໃຫ້ໂນດຮາກຢ່າງເຫິງສຸດ ຖ້າມີໂນດຢ່າງເຫິງຂຶ້ນໄປ 1 ໂນດ ໂນດນັ້ນເອີ້ນວ່າ ໂນດແມ່ (Parent Node) ໃນທາງດູງວັກນ ໂນດທີ່ຢູ່ລຸ່ມມັນເອີ້ນວ່າ ໂນດລູກ (Child Node) ແລະ ໂນດທັງໝົດທີ່ແຕກອອກມາຈາກໂນດແມ່ດູງວັກນເອີ້ນວ່າ ໂນດພິນ້ອງ (Sibling Node). ຖ້າມີເສັ້ນທາງຈາກ X ໄປຫາໂນດ Y ເຮົາຈະເອີ້ນ X ວ່າເປັນ ໂນດເທິງ (Ancestor Node) ຂອງ Y ແລະ Y ເປັນ ໂນດລູ່ມ (Descendant Node) ຂອງ X ສໍາລັບໂນດອື່ນໆ ທີ່ເປັນຮາກຂອງຕົ້ນໄມ້ຍ່ອຍ (Subtree) ຈະປະກອບດ້ວຍຕົວຂອງມັນເອງ ແລະ ໂນດລຸ່ມທັງໝົດທີ່ຢູ່ກ້ອງໂຕມັນ.

ຖ້າກຳນົດສັນຍາລັກ G ແທນຄວາມໝາຍຂອງກຣາບ ອົງປະກອບທີ່ສໍາຄັນຂອງ G ຈະປະກອບດ້ວຍກຸ່ມ ຂອງໂນດ ເຊິ່ງແທນດ້ວຍສັນຍາລັກ s ແລະກຸ່ມຂອງເສັ້ນເຊື່ອມ ເຊິ່ງແທນດ້ວຍສັນຍາລັກ E ແລະ ກຣາບ G ຈະແທນດ້ວຍ G (S, E) ຖ້າ $\{s_1, s_2, s_3, \dots\}$ ຄືໂນດໃນກຸ່ມ s ແລະ $\{e_1, e_2, e_3, \dots\}$ ຄືເສັ້ນເຊື່ອມໃນກຸ່ມ e ເຮົາສາມາດຊູນຄວາມສໍາພັນຂອງເສັ້ນເຊື່ອມແລະໂນດໄດ້ດັ່ງນີ້ $e_1 = (s_1, s_2)$ ເຊິ່ງໝາຍເຖິງ e_1 ຄືເສັ້ນເຊື່ອມຕໍ່ລະຫວ່າງໂນດ s_1 ແລະ s_2 ແລະ s_1 ແລະ s_2 ຈະເອີ້ນວ່າ ໂນດຂ້າງຄົງ (Adjacent Node) ດັ່ງຮູບທີ 2.2



ຮູບທີ 2.2 Node ແລະ ເສັ້ນເຊື່ອມຂອງ ກຣາບ G (S, E)

ຈຳນວນເສັ້ນທີ່ຕໍ່ມາຫາໂນດຈະເອີ້ນວ່າ ດີກຮີ (Degree) ຕົວຢ່າງ ໂນດ s_1 ຂອງກຣາບ G(S,E) ໃນຮູບທີ 2.2 ຈະມີເສັ້ນເຊື່ອມ e_1 ແລະ e_2 ຕໍ່ມາຮອດໂຕມັນ, ດັ່ງນັ້ນ ດີກຮີ ຂອງ s_1 ຈະເທົ່າກັບ 2 ເຊິ່ງເຮົາສາມາດຊູນເປັນສັນຍາລັກໄດ້ແບບນີ້ $\deg(s_1) = 2$. ແຕ່ຖ້າໂນດໄດ້ມີດີກຮີເປັນ 0 ເອີ້ນວ່າ ໂນດດູ່ວ (Isolated Node).

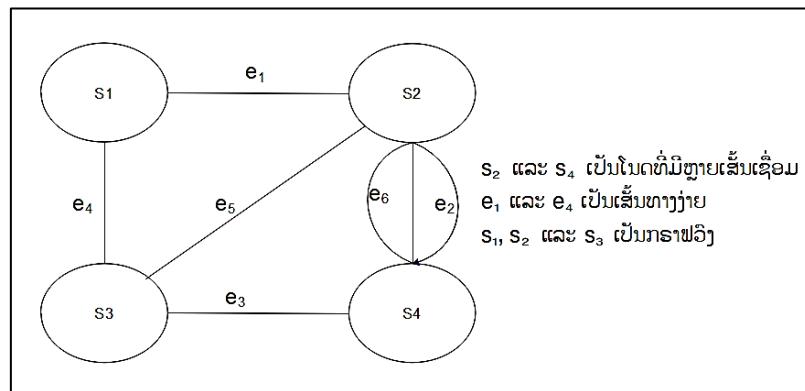
ຖ້າໃຫ້ p ໝາຍເຖິງເສັ້ນທາງຈາກໂນດ 1 ໄປຫາໂນດ 1 ຄວາມຍາວຂອງ p ຄືຈຳນວນເສັ້ນເຊື່ອມ e ທີ່ເຊື່ອມຢ່າງຫວ່າງທາງໄປຂອງໂນດ 2 ໂນດທີ່ກຳລັງພິຈາລະນາ ຫຼືເທົ່າກັບຈຳນວນໂນດລົບດ້ວຍ 1 ແລະເສັ້ນທາງສາມາດຊູນແທນດ້ວຍສັນຍາລັກຕໍ່ໄປນີ້:

$$p = (s_0, s_1, s_2, s_3, \dots, s_n)$$

ໝາຍເຖິງທາງໄປຈາກໂນດ s_0 ຫາໂນດ s_n ມີຄວາມຍາວເທົ່າກັບ n (ຈຳນວນໂນດທັງໝົດຄື $n + 1$ ໂນດ) ແລະຈາກເສັ້ນທາງ p ຫາໂນດ s_0 ແລະ s_n ຄືໂນດດູງວັກນຈະເອີ້ນວ່າ ເສັ້ນທາງປິດ (Close

Path), ແຕ່ຖົກໂນດທຸກໂນດຈາກ s_0, \dots, s_n ບໍ່ຊຳກັນເສັ້ນຫາງນີ້ເອີ້ນວ່າ *ເສັ້ນຫາງງ່າຍ (Simple Path)* ແລະ ຖົກເສັ້ນຫາງງ່າຍແບບປິດມີຄວາມຍາວຕັ້ງແຕ່ 3 ຂຶ້ນໄປເອີ້ນວ່າ *ກຮາບວິງ (Cycle Graph)*.

ກຮາບຈະຖືກເອີ້ນວ່າ *ກຮາບເຊື່ອມ (Connected Graph)* ຖົກທຸກໂນດໃນກຮາບມີເສັ້ນຫາງງ່າຍຕໍ່ເຖິງກັນໝົດ ແລະ ກຮາບຈະຖືກເອີ້ນວ່າ *ກຮາບສົມບູນ (Complete Graph)* ຖົກທຸກໂນດໃນກຮາບເປັນໂນດຂ້າງຄົງກັນໝົດ ແລະ ໃນກຮາບເຊື່ອມຕໍ່ຖົກບໍ່ມີລູບເກີດຂຶ້ນຄືບມີເສັ້ນຫາງປິດກຮາບນັ້ນຈະເປັນຕົ້ນໄມ້.



ຮູບ 2.3 ລັກສະນະຂອງເສັ້ນຫາງໃນກຮາບ

ຖົກເສັ້ນເຊື່ອມໃນກຮາບມີການລະບຸຂໍ້ມູນຈະເອີ້ນວ່າ *ກຮາບບ້າຍ (Labelled Graph)* ແລະ ໃນກໍລະນີຂໍ້ມູນທີ່ລະບຸເປັນຕົວເລກທີ່ບອກຂະໜາດຂອງຄວາມສໍາຄັນຂອງເສັ້ນເຊື່ອມຈະເອີ້ນວ່າ ນີ້ໜີ້ກ (Weight) ໃນກໍລະນີທີ່ໂນດຄູ່ງວັນ, ແຕ່ມີເສັ້ນເຊື່ອມຕໍ່ກັນຢູ່ໝາຍກ່າວໜຶ່ງເສັ້ນເຊື່ອມຈະເອີ້ນເສັ້ນເຊື່ອມເງື່ອງນັ້ນວ່າ *ເສັ້ນເຊື່ອມຫຼາຍ (Multiple Edges)* ເຊິ່ງເສັ້ນເຊື່ອມດ້ວຍກ່າວສາມາດຂຽນໄດ້ດ້ວຍນີ້:

$$e_1 (s_1, s_2) \text{ ແລະ } e_2 (s_1, s_2)$$

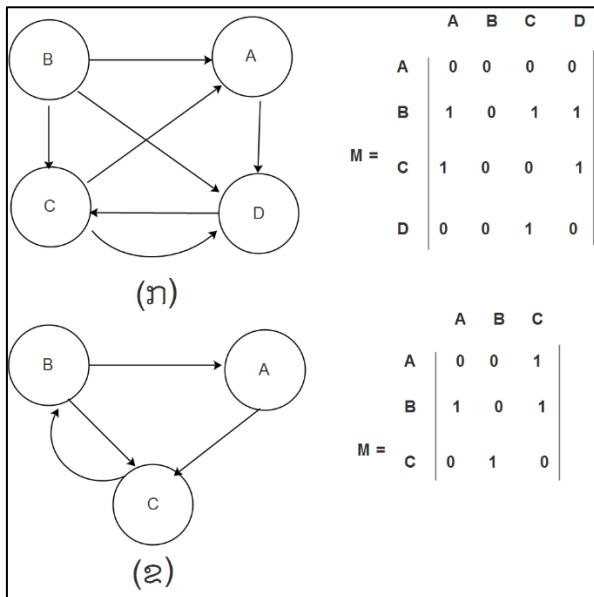
ຖົກເສັ້ນເຊື່ອມທີ່ເກີດຂຶ້ນໃນກຮາບມີການກຳນົດທິດທາງເຊັ່ນ ເລີ່ມຈາກໂນດ s_1 ໄປຫາ s_2 ກຮາບນັ້ນຈະເອີ້ນວ່າ *ກຮາບມີທິດ (Directed Graph)* ໃນການສະແດງຮູບແບບຂອງກຮາບເພື່ອນຳມາໃຊ້ປະໂຫຍດນັ້ນມີຢູ່ໝາຍຮູບແບບ ແຕ່ຮູບແບບທີ່ນີ້ຍືມກັນຫຼາຍຄືການໃຊ້ມາຕຣິດເອີ້ນວ່າ ມາຕຣິດແບບໃກ້ກັນ (Adjacency Matrix) ມາຕຣິດແບບໃກ້ກັນຄືມາຕຣິດທີ່ສັງຈາກຮາບ ໂດຍກຳນົດວ່າ ຖົກ a_{ij} ເປັນອົງປະກອບ (Element) ໃນມາຕຣິດ A ຄ່າຂອງ a_{ij} ຈະເປັນດັ່ງນີ້

a_{ij} ຈະເທົ່າກັບ 1 ຖົກ s_i ເປັນໂນດຂ້າງຄົງກັບ s_j

a_{ij} ຈະເທົ່າກັບ 0 ຖົກ s_i ແລະ s_j ບໍ່ເປັນໂນດຂ້າງຄົງກັນ

	e_1	e_2	e_3	...	e_j
s_1	a_{11}	a_{12}	a_{13}	...	a_{1j}
s_2	a_{21}	a_{22}	a_{23}	...	a_{2j}
$M =$	a_{31}	a_{32}	a_{33}	...	a_{3j}
\vdots	\vdots	\vdots	\vdots		\vdots
s_i	a_{i1}	a_{i2}	a_{i3}	...	a_{ij}

ຮູບທີ 2.4 ມາຕຣິດແບບໃກ້ກັນໃຊ້ແນນການກາບ



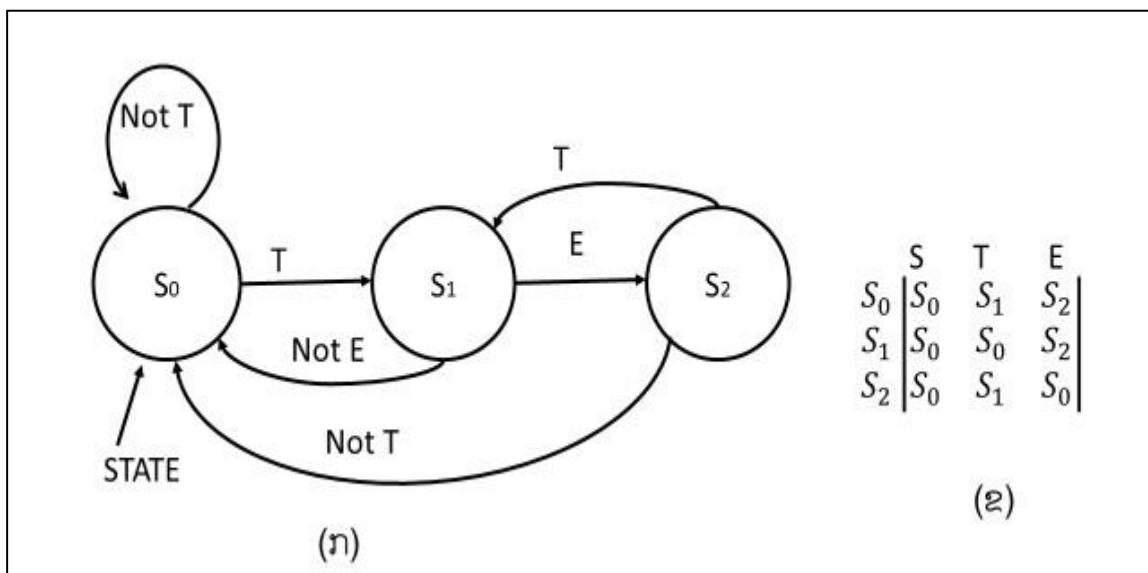
ຮູບທີ 2.5 ມາຕຣິດແບບໃກ້ກັນຂອງການກາບ

2.1.2 ເຄືອງສະຖານະຈຳກັດ (Finite State Machine)

ເຄືອງສະຖານະຈຳກັດ ຫຼື ເຄືອງອັດຕະໂນມັດຈຳກັດ (Finite Automation) ເປັນວິທີການອະຫິບາຍຮູບແບບຂອງພິດຕິກຳ ເຊັ່ນ: ການປິດ-ເປີດປະຕູລືບ ແລະ ການເຮັດວຽກຂອງໂຕແບ່ງສ່ວນ (Parser) ເປັນຕົ້ນ, ທີ່ຂົງແນນດ້ວຍການເຊື່ອມຕໍ່ຊະນິດການກຳມືດ. ເຄືອງສະຖານະຈຳກັດມີອີງປະກອບ 3 ຢ່າງຄື: ຊຸດຂອງສະຖານະ ຫຼື ໂນດ, ຊຸດຂອງຄ່ານຳເຂົ້າ (Input Values) ແລະ ພັງຊັນການປົງປານສະຖານະ (State Transition Function), ການເຮັດວຽກຂອງເຄືອງສະຖານະຈຳກັດຈະເລີ່ມຕົ້ນ (S_0) ແຊິ່ງເຫັນວ່າສະຖານະບັດຈຸບັນ, ເມື່ອມີຄ່ານຳເຂົ້າເຮັດວຽກກັບສະຖານະບັດຈຸບັນ ສະຖານະນີ້ກໍຈະປົງປານໄປຕາມເງື່ອນໄຂຂອງພັງຊັນການປົງປານສະຖານະ, ແຊິ່ງອາດຈະປົງປານສະຖານະໃໝ່ ຫຼື ບໍ່ປົງປານ

ສະຖານະ. ເມື່ອມີຄ່ານໍາເຂົາຊຸດໃຫມ່ເຂົ້າມາເຮັດວຽກໃນສະຖານະປັດຈຸບັນມີອີກ, ການປ່ຽນແປງກໍຈະເກີດຕາມເງື່ອນໄຂຂອງຟັງຊັນການປ່ຽນສະຖານະອີກ ແລະ ເຮັດວິນໄປແບບມີເລື່ອຍໆຈົນຈົບຂະບວນການ.

ເຄື່ອງສະຖານະຈຳກັດຊະນິດ ຕົວຮັບ (*Acceptor*) ເປັນເຄື່ອງສະຖານະຈຳກັດແບບງ່າຍຊະນິດໜຶ່ງ, ທີ່ຕອບຮັບ ຫຼື ປະຕິເສດຕໍ່ຄ່ານໍາເຂົາເທົ່ານັ້ນ, ຖ້າຄ່ານໍາເຂົາທັງໝົດທີ່ມີຕໍ່ສະຖານະປັດຈຸບັນໄດ້ຮັບການຕອບຮັບ, ຕ່ານໍາເຂົາກໍຈະໄດ້ຮັບການຍອມຮັບ ແລະ ຈະປ່ຽນສະຖານະປັດຈຸບັນໄປຢູ່ສະຖານະທັດໄປ. ຖ້າສະຖານະປັດຈຸບັນຕອບປະຕິເສດ, ຄ່ານໍາເຂົານັ້ນກໍຈະຖືກປະຕິເສດຄືກັນ. ຕົວຢ່າງຂອງເຄື່ອງສະຖານະຈຳກັດຊະນິດຕົວຮັບ ທີ່ມີການຄົ້ນຫາຕົວອັກສອນໃນຄໍາ, ຖ້າເຮົາຈະຫາຕົວອັກສອນວ່າ “TE” ຈາກຄໍາ “STATE” ລັກສະນະຂອງເຄື່ອງຈະເປັນດັງສະແດງໃນຮູບ 2.6



ຮູບ 2.6 ເຄື່ອງສະຖານະຈຳກັດຊະນິດຕົວຮັບ

ຈາກຮູບ 2.6 ໂນດຕ່າງໆໃນການບະຈາສະແດງສະຖານະຂອງເຄື່ອງສະຖານະຈຳກັດ, ເຊິ່ງມີ s_0 , s_1 ແລະ s_2 ເປັນສະຖານະ. ຄ່ານໍາເຂົາຂອງລະບົບຄື “STATE” ເຊິ່ງປະກອບດ້ວຍຕົວອັກສອນ 5 ຕົວ, ພັງຊັນການປ່ຽນສະຖານະຄືຕົວອັກສອນ “T” ສໍາລັບສະຖານະ s_0 ແລະ “E” ສໍາລັບສະຖານະ s_1 ການເຮັດວຽກຂອງເຄື່ອງເລີ່ມຕົ້ນຈາກຄ່ານໍາເຂົາ “STATE” ຕົວອັນສອນແຕ່ລະຕົວຂອງຄໍາຈະຖືກສົ່ງເຂົ້າໄປໃນສະຖານະ s_0 ເພື່ອລະຕົວ, ຈາກຕົວທໍາອິດຫາຄົວສຸດທ້າຍຂອງຄໍາ. ຖ້າຕົວອັກສອນທີ່ຖືກສົ່ງເຂົ້າໄປບໍ່ແມ່ນ “T” ສະຖານະ s_0 ກໍຈະຢູ່ຄົງທີ່, ແຕ່ຖ້າແມ່ນຈະຍັງໄປສະຖານະໃໝ່ ຄື s_1 ແລະ ກວດສອບຕົວອັກສອນຕໍ່ໄປ. ຖ້າຄ່ານໍາເຂົາຕໍ່ໄປເປັນ “E” ການກວດສອບກໍຈະສຳເລັດ, ແຕ່ຖ້າວ່າຄ່ານໍາເຂົາຕົວໃຫມ່ບໍ່ແມ່ນ “E” ສະຖານະໃຫມ່ຈະປ່ຽນໄປຢູ່ s_0 ແລະ ເລີ່ມຕົ້ນໃຫມ່, ຈາກນັ້ນຕົວອັກສອນໃຫມ່ຈະຖືກປ້ອນເຂົ້າມາຈົນຈົບຕົວອັກສອນສຸດທ້າຍຂອງຄໍາ.

ຈາກຕົວຢ່າງຂ້າງເທິງ, ການເຮັດວຽກແບບລະອງດຕະເລີ່ມຈາກຕົວອັກສອນທຳອິດ “S” ຖືກສົ່ງເຂົ້າມາທີ່ s_0 ເມື່ອຈາກ s_0 ບໍ່ຄັບພັງຊັນການປ່ຽນສະຖານະ, ດັ່ງນັ້ນຕົວອັກສອນ “S” ຈະຖືກປະຕິເສດ ແລະ ສະຖານະປັດຈຸບັນຢູ່ທີ່ s_0 , ຈາກນັ້ນຕົວອັກສອນໃໝ່ຈະຖືກສົ່ງເຂົ້າມາທີ່ສະຖານະ s_0 ຄື “T”

ແລະ ໄດ້ຮັບການຕອບຮັບເຮັດໃຫ້ສະຖານະປັດຈຸບັນຍ້າຍໄປເປັນສະຖານະໃໝ່ s_1 ແລະ ຕົວອັນສອນໃໝ່ ຈະຖືກສິງເຂົາມາໃນສະຖານະ s_1 ຄື “A” ເນື່ອງຈາກພັງຊັນການປ່ຽນສະຖານະຄື “E” ເຮັດໃຫ້ສະຖານະໃໝ່ຂອງສະຖານະປັດຈຸບັນຍ້າຍໄປທີ່ s_0 ຄືນອີກ ແລະ ຕົວອັນສອນໃໝ່ “T” ກໍຈະຜ່ານຈາກ s_0 ໄປ s_1 , ຈາກນັ້ນ “E” ກໍຈະເຂົາມາເຮັດໃຫ້ສະຖານະຍ້າຍຈາກ s_1 ໄປ s_2 ແລະສິນສຸດການເຮັດວຽກ.

ການປ່ຽນສະຖານະເຄື່ອງສະຖານະຈຳກັດສາມາດຂຽນໃນຮູບແບບຂອງມາຕຣິດໄດ້ດັ່ງສະແດງໃນຮູບ 2.6 (ຂ) ເຊິ່ງເອີ້ນວ່າ ທານຊື່ເຂັ້ມາຕຣິດ (*Transition Matrix*).

ນິຍາມທາງຄະນິດສາດຂອງເຄື່ອງສະຖານະຈຳກັດສາມາດຂຽນໄດ້ດັ່ງນີ້:

ເຄື່ອງສະຖານະຈຳກັດຊະນິດຕົວຮັບ ສັນຍາລັກ 4 ຕົວຄື: (Σ, S, S_0, Δ) ເຊິ່ງ

- Σ ຄືຄ່ານໍາເຂົ້າ
- S ເປັນກຸ່ມຂອງສະຖານະ
- S_0 ເປັນສະຖານະຕ່າງໆໃນກຸ່ມຂອງ S
- Δ ເປັນພັງຊັນການປ່ຽນສະຖານະເທົ່າກັບ $\Delta : S * \Sigma \rightarrow S$

2.1.3 ການກຳນົດນິຍາມໃຫ້ກັບບັນນຫາ

ເພື່ອເປັນການທຳຄວາມເຂົ້າໃຈໃນເຕັກນິກຂອງບັນຍາປະດິດເບື້ອງຕົ້ນ ໃນທີ່ນີ້ຈະສະແດງໃຫ້ເຫັນເຖິງວິທີການແກ້ບັນຫາ ແລະຫຼັກການບາງຢ່າງໃນການກຳນົດຫຼັກເກມຂອງບັນຫາ ວິທີການແລະຫຼັກການດັ່ງກ່າວເປັນຂະບວນການພື້ນຖານທີ່ຈະທຳໃຫ້ເຂົ້າໃຈການແກ້ບັນຫາ (problem solving) ທາງດ້ານຄອມພິວເຕີ ກຳນົດໃຫ້ມືໂຖ 2 ໃບທີ່ມີຂະໜາດໃສ່ນັ້າ 4 ແລະ 3 ລົດຕາມລຳດັບ ຈະມີວິທີການແນວໃດເພື່ອເຮັດໃຫ້ໂຖນັ້ນທີ່ມີຂະໜາດ 4 ລົດ ມິນ້າຢູ່ 2 ລົດຝຶດີ ໂດຍທີ່ບໍ່ອາໄສເຄື່ອງວັດແທກໄດ້ຈາກໂຈດດັ່ງກ່າວຂ້າງຕົ້ນ ການແກ້ບັນຫາມີໝາຍວິທີ ແຕ່ໃນທີ່ນີ້ຈະອາໄສວິທີການຂອງ ຂອບເຂົ້າສະຖານະ (state space) ມາຊ່ວຍໃນການແກ້ ໂດຍທີ່ກຳນົດຈຸດເລີ່ມຕົ້ນຂອງບັນຫາ ແລະ ເປົ້າໝາຍຂອງການແກ້ບັນຫາ ວ່າຄືຫຍັງ ສຳລັບເລື່ອງໂຖນັ້າ ການແກ້ບັນຫາຈະເລີ່ມຕົ້ນຈາກ ໂຖນັ້າ 2 ໂຕທີ່ບໍ່ມິນັ້າເລີຍ ແລະ ຜົນຮັບສຸດທ້າຍ ຈະຕົ້ອງເປັນ ມິນັ້າ 2 ລົດໃນໂຖທຳອິດ ແລະ ໂຖທີ່ 2 ມິນັ້າທີ່ໄດ້ກໍໄດ້. ຖ້າກຳນົດວ່າ X ເຊິ່ງເປັນຕົວປ່ຽນໄດ້ ໚ໍ່ທີ່ແກນຄ່າຂອງປະລິມານນັ້າໃນໂຖນັ້າທຳອິດ ແລະ Y ຄືຕົວປ່ຽນໄດ້ ໚ໍ່ທີ່ແກນຄ່າຂອງປະລິມານນັ້າໃນໂຖນັ້າທີ່ 2 ດັ່ງນັ້ນຄ່າຕ່າງໆ ທີ່ເປັນໄປໄດ້ທັງໝົດຂອງ X ແລະ Y ຈະເປັນດັ່ງນີ້

$X = 0, 1, 2, 3, 4$ (ເນື່ອງຈາກໂຖນັ້າທຳອິດບັນຈຸນັ້າໄດ້ 4 ລົດ) ແລະ

$Y = 0, 1, 2, 3$ (ເນື່ອງຈາກໂຖນັ້າທີ່ສອງບັນຈຸນັ້າໄດ້ 3 ລົດ)

ແລະປະລິມານນັ້າໃນໂຖນັ້າ 2 ໂຖ ສາມາດຂຽນຢູ່ໃນຮູບຂອງ X ແລະ Y ໄດ້ເປັນ (X, Y) ຈຸດເລີ່ມຕົ້ນໃນການແກ້ບັນຫາເລີ່ມຈາກ $(0, 0)$ ເຊິ່ງໝາຍເຖິງສະຖານະທີ່ບໍ່ມິນັ້າໃນໂຖນັ້າ ສຳລັບເປົ້າໝາຍທີ່ຕົ້ອງການຄື $(2, n)$ ໂດຍ n ເປັນເລກໄດ້ ທີ່ມີຄ່ານ້ອຍກວ່າ 3ໃນການແກ້ບັນຫາ ຈະຕົ້ອງທຳການປ່ຽນສະຖານະ ຈາກ $(0, 0)$ ໄປເປັນ $(2, n)$ ເຊິ່ງຈະເຮັດໄດ້ໂດຍການອາໄສກົດຕ່າງໆທີ່ກ່ຽວກັບການເຫັນຂອງບັນຫານີ້, ກົດເຫຼົ່ານີ້ໄດ້ແກ່

1. ຕົມນັ້າລົງໃນໂຖທຳອິດຈົນເຕັມ

2. ເຕີມນໍາລົງໃນໄຕໃບສອງຈິນເຕັມ
3. ເຫນົ້າບາງສ່ວນອອກຈາກໄຕທຳອິດ
4. ເຫນົ້າບາງສ່ວນອອກຈາກໄຕສອງ

ຖ້າທາກອະທິບາຍກົດເງື່ອນີ້ໃນຮູບຂອງຕົວປັບປຸງ X ແລະ Y ຈະໄດ້ດັ່ງຕາຕະລາງທີ 2.1

ລຳດັບ	ກົດເກີນ	ຄໍາອະທິບາຍ
1	(X,Y: $X < 4$) -> (4,Y)	ຖອກນັ້າໃສ່ໄຕທຳອິດຈິນເຕັມ
2	(X,Y: $Y < 3$) -> (X,3)	ຖອກນັ້າໃສ່ໄຕທີ 2 ຈິນເຕັມ
3	(X,Y: $X > D$) -> (X-D,Y)	ຖອກນັ້າບາງສ່ວນອອກຈາກໄຕ 2
4	(X,Y: $Y > D$) -> (X,Y-D)	ຖອກນັ້າອອກຈາກໄຕທຳອິດຈິນໝີດ
5	(X,Y: $Y > 0$) -> (X,0)	ຖອກນັ້າອອກຈາກໄຕທີ 2 ຈິນໝີດ
6	(X,Y: $X+Y \geq 4 \wedge Y > 0$) -> (4, $Y-(4-X)$)	ຖອກນັ້າອອກຈາກໄຕທີ 2 ໃສ່ໄຕທຳອິດ ຈິນເຕັມ ໂດຍໄຕທີ 1 ມີນໍາລວມກັນ ໜ້າຍກວ່າ 4 ລຶດ
7	(X,Y: $X+Y \geq 3 \wedge X > 0$) -> ($X-(3-Y),3$)	ຖອກນັ້າອອກຈາກໄຕທຳອິດ ໃສ່ໄຕທີ 2 ຈິນເຕັມ ໂດຍ 2 ໄຕ ມີນໍາລວມກັນໜ້າຍ ກວ່າ 3 ລຶດ
8	(X,Y: $X+Y \leq 4 \wedge Y > 0$) -> ($X+Y,0$)	ຖອກນັ້າອອກຈາກໄຕທີ 2 ໃສ່ໄຕທຳອິດ ຈິນເຕັມ ໂດຍ 2 ໄຕ ມີນໍາລວມກັນໜ້າຍ ກວ່າ 4 ລຶດ
9	(X,Y: $X+Y \leq 3 \wedge X > 0$) -> (0, $X+Y$)	ຖອກນັ້າອອກຈາກໄຕທີ 2 ໃສ່ໄຕທີ 1
10	($X > 0, X+Y \leq 3$) → (0, $X+Y$)	ຖອກນັ້າອອກຈາກໄຕທຳອິດ ໃສ່ໄຕທີ 2

ຕາຕະລາງທີ 2.1 ກົດທີ່ອະທິບາຍການແກ້ບັນຫາເລື່ອງໄຕນີ້

ໃນການກຳນົດນິຍາມຂອງບັນຫານັ້ນມີສິ່ງທີ່ຄວນເອົາໃຈໃສດັ່ງນີ້

1. ໃນການຂຽນກົດເພື່ອອະທິບາຍບັນຫານີ້ຈະຕ້ອງມີອີງປະກອບພື້ນຖານຢ່າງໜ້ອຍ 2 ສ່ວນທີ່ໃຊ້ກວດສອບເງື່ອນໄຂ (condition) ແລະ ຂໍສະຫຼຸບ (conclusion). ສ່ວນທີ່ເປັນເງື່ອນໄຂຄືສ່ວນທີ່ໃຊ້ກວດສອບເພື່ອນຳກົດຂໍ້ນັ້ນມາໃຊ້ ແລະ ສ່ວນທີ່ເປັນຂໍສະຫຼຸບແມ່ນສ່ວນທີ່ນຳໃຊ້ກົດ ເຊັ່ນ:

$$(X < 4, Y) \rightarrow (4, Y)$$

ໝາຍຄວາມວ່າ ຖ້າ X ນອຍກວ່າ 4 ລົດ ($X < 4, Y$) ຈະສະຫຼຸບວ່າເຕີມາດເຮັດໃຫ້ມິນໍາຢູ່ 4 ລົດໃນໄຕ່ມິນໍາທຳອິດໄດ້ ແລະມິນໍາຈຳນວນ Y ໃນໄຕ່ມິນໍາທີ່ສອງຜົນທີ່ໄດ້ອອກມາຈະເປັນ $(4, y)$

ລັກສະນະຂອງກົດທີ່ສະແດງຂ້າງເທິງ ເຮັດວ່າ ກົດການຜະລິດ (*production rule*) ເຊິ່ງກົດການຜະລິດນີ້ຈະຕ້ອງ

ກຳນົດຫຼືສ້າງກົດຈຳນວນໜີ້ທີ່ສາມາດອະທິບາຍການເຮັດວຽກທັງໝົດທີ່ຈຳເປັນຕໍ່ການແກ້ປັນຫານັ້ນ ພ

2. ການກຳນົດຄ່າຂອງ X ແລະ Y ຈະຕ້ອງສາມາດຄອບຄຸມເຖິງຄວາມເປັນໄປໄດ້ຫັງໝົດ ເຊິ່ງຄ່າຂອງ X

ແລະ Y ມີຈະເປັນຕົວໃນການກຳນົດສະຖານະ (state) ຕ່າງໆ ຂອງການແກ້ປັນຫາ ໃນທີ່ນີ້ເຮົາກຳນົດສະຖານະຕ່າງໆ ເປັນ

(X,Y) ໄດຍມີຄ່າ $X = 0, 1, 2, 3, 4$ ແລະ $y = 0, 1, 2, 3$

3. ในกรณีมีดัชนีตอนต่างๆ ของงานแก้ปั๊มห้าເຮັດວ່າ ຂອບເຂດສະຖານະ (state space), ເຊິ່ງກຳນົດວ່າຂອບເຂດ(space) ຫີ້ງໆ ຢ່າງນ້ອຍຈະຕ້ອງມີ 2 ສະຖານະຄື ສະຖານະເລີ່ມຕົ້ນ (Start State) ເປັນ $(0,0)$ ແລະ ສະຖານະເປົ້າໝາຍ(Goal State) ເປັນ $(2,n)$ ການແກ້ປັ້ນຫາເປັນການນຳກິດ ມາ ອະທິບາຍການປຸ່ງແປງຕ່າງໆ ຂອງສະຖານະ ຈາກສະຖານະເລີ່ມຕົ້ນຈຶນເຖິງສະຖານະເປົ້າໝາຍ ໃນການແກ້ປັ້ນຫາຕາມລັກສະນະນີ້ ຈະເຮັດວ່າ ລະບົບການຜະລິດ (production system). ຈາກຕົວຢ່າງຂ້າງຕົ້ນ ສະຖານະຂອງປັ້ນຫາເລີ່ມຕົ້ນຄື $(0,0)$ ເຮົາຈະຕ້ອງຫາທາງປຸ່ງຈາກ $(0,0)$ ໃຫ້ເປັນ $(2,n)$ ການປຸ່ງເລີ່ມຈາກການສໍາໜູວດກິດ ຈາກຂີ້ 1 ເຖິງ 8 ຂ້າງເທິງ ວ່າມີກິດຂີ້ໄດ້ສາມາດໃຊ້ກັບສະຖານະເລີ່ມຕົ້ນໄດ້

1. $(X < 4, Y) \rightarrow (4, Y)$
 2. $(X, Y < 3) \rightarrow (X, 3)$
 3. $(X > 0, Y) \rightarrow (0, Y)$
 4. $(X, Y > 0) \rightarrow (X, 0)$
 5. $(X + Y \geq 4, Y > 0) \rightarrow (4, Y - (4 - X))$
 6. $(X > 0, X + Y \geq 3) \rightarrow (X - (3 - Y), 3)$
 7. $(X + Y \leq 4, Y > 0) \rightarrow (X = Y, 0)$
 8. $(X > 0, X + Y \leq 3) \rightarrow (0, X + Y)$

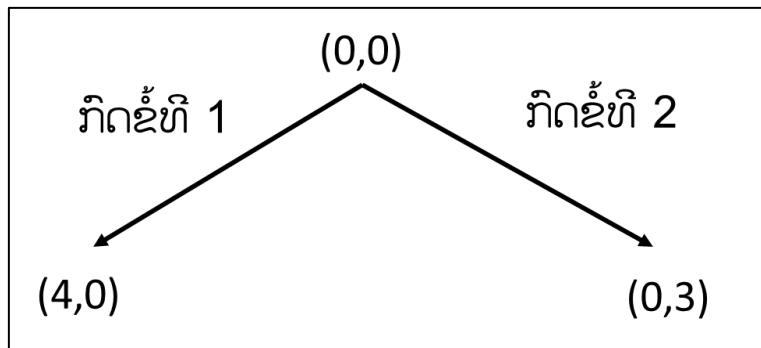
ຈາກການສໍາຫຼວດມີກິດຂໍ້ 1 ແລະ 2 ເທົ່ານັ້ນທີ່ສາມາດນຳມາໃຊ້ກັບເງື່ອນໄຂນີ້ໄດ້ ເພະວ່າໃນສ່ວນຂອງເງື່ອນໄຂ

គិត $(X < 4, Y)$ និង $(X, Y < 3)$ តាមលំដាប់ បែងចុះតាមសម្រាប់លើកទី ១ និងទី ២ តាមរូបខាងក្រោម

ខែងភីដខ្លះ 1 និង $Y < 3$ ពាមពេជ្រើន ខែងភីដខ្លះ 2 តំងមូលជាកសាងទាមនេះ $(0,0)$ និង $(4,0)$

ສາມາດປ່ຽນຈາກ

(0,0) ເປັນ (0,3) ໄດ້ໄດຍກົດຂໍທີ 2 ເຊິ່ງສາມາດຂຽນເປັນຮູບໄດ້ຕໍ່ນີ້



ຮູບທີ 2.7 ການປ່ຽນສະຖານະຈາກ (0,0) ເປັນ (4,0) ແລະ (0,3)

ຈາກນັ້ນສ້າງການສໍາຫຼວດສະຖານະ (4,0) ກໍພືບວ່າມີກົດຂໍ 2, 3 ແລະ 6 ທີ່ມີເງື່ອນໄຂກົງກັນ ແລະ ສະຖານະຕ່າງໆ ຈະປ່ຽນໄປຕໍ່ນີ້

(4,0) -> (4,3) ໂດຍການໃຊ້ກົດຂໍ 2

(4,0) -> (0,0) ໂດຍການໃຊ້ກົດຂໍ 3

(4,0) -> (1,3) ໂດຍການໃຊ້ກົດຂໍ 6

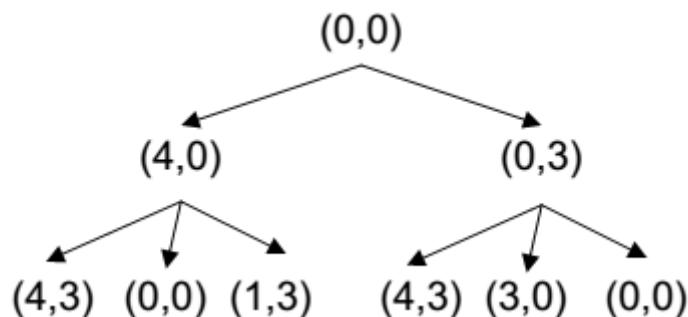
ຈາກການສໍາຫຼວດສະຖານະ (0,3) ມີກົດຂໍ 1, 4 ແລະ 7 ທີ່ມີເງື່ອນໄຂກົງກັນ ແລະສະຖານະຕ່າງໆ ທີ່

ເກີດໃໝ່ຈະໄດ້ເປັນຕໍ່ນີ້

(0,3) -> (4,3) ໂດຍກົດຂໍ 1

(0,3) -> (0,0) ໂດຍກົດຂໍ 4

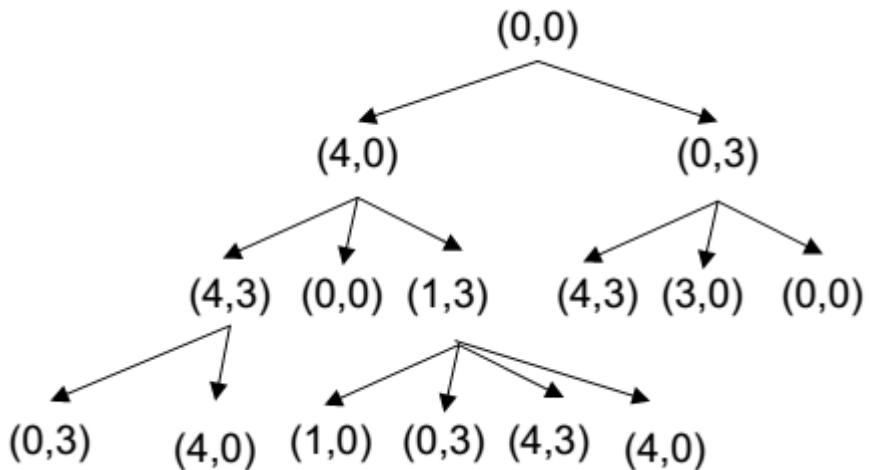
(0,3) -> (3,0)



ຮູບທີ 2.8 ການຂະຫຍາຍໂນດມາໃນລະດັບທີສອງ

ຈາກນັ້ນໄປຂະຫຍາຍໂນດ (4,3) ທີ່ຢູ່ແຖວລຸ່ມສຸດແລະຢູ່ຊ້າຍມີສຸດຕາມຮູບທີ 2.8 ແລ້ວຫາກົດທີ່ທີ່ເໝາະສົມມາໃຊ້ແລ້ວຂ້າມໄປພິຈາລະນາ (1,3) ແລະ (3,0) ຕາມລຳດັບ ເຊິ່ງຈະເຫັນໄດ້ວ່າໂນດ

(0,0) (4,3) และ (0,0) จะบ่งชี้ว่าเป็นโหนดที่เข้ากับโหนดที่มีปู่แล้ว เมื่อพิจารณาจินติ์
และ บ่งชี้ว่าเป็นโหนดเดียว เนื่องจากได้โหนดห่างมิดตั้งนี้



ຮູບທີ 2.9 ຂອບເຂດປັນຫາຂອງການແກ້ປັນຫາ

ຈາກຕົ້ນໄມ້ຂອງຮູບຂ້າງເທິງ ຈະເຫັນວ່າມີຢູ່ຫຼາຍເສັນຫາງທີ່ສາມາດແກ້ປັນຫານີ້ໄດ້ ຖ້າເຖິງຈະເລື່ອກ
ເສັນຫາງໜຶ່ງຂອງການແກ້ປັນຫາສາມາດຊົ່ມເປັນຕາຕະລາງຂອງການແກ້ປັນຫາໄດ້ດັ່ງນີ້

ໂທ 1	ໂທ 2	ກົດຂໍ້ທີ	ສະຖານະ
0	0		ເລີ່ມຕົ້ນ
0	3	2	
3	0	7	
3	3	2	
4	2	5	
0	2	3	
2	0	7	ເປົ້າໝາຍ

ຮູບທີ 2.9 ສະແດງການແກ້ປັນຫາເລື່ອງໂທນີ້

ການແກ້ປັນຫາແບບທີ່ກ່າວມາແລ້ວເປັນຕົວຢ່າງຂອງການແກ້ປັນຫາແບບ ລະບົບການຜະລິດ
(Production) ເຊິ່ງເປັນຂະບວນການໃນການວາງໂຄງສ້າງໂປຣແກຣມແບບປັນຍາປະດິດ ໃຫ້ມີລັກສະນະ
ງ່າຍຕໍ່ການອະທິບາຍ ຂະບວນການ (ຫຼື ວິທີການແກ້ປັນຫາ) ການແກ້ປັນຫາແບບນີ້ຈະຕ້ອງປະກອບໄປ
ດ້ວຍ

1. ກົດເຊິ່ງກົດແຕ່ລະຂໍຈະຕັ້ງປະກອບດ້ວຍສ່ວນທີ່ຢູ່ທາງຊ້າຍທີ່ຈະອະທິບາຍເຖິງເງື່ອນໄຂຂອງກົດ ແລະ ສ່ວນທີ່ຢູ່ທາງຂວາທີ່ອະທິບາຍເຖິງຜົນຂອງກົດ
 2. ຖານຂໍ້ມູນທີ່ມີການສະຫຼຸບທີ່ຕັ້ງການ ບາງສ່ວນຂອງຖານຂໍ້ມູນຈະເປັນແບບຖາວອນ ແລະບາງສ່ວນຈະເປັນສິ່ງທີ່ວ່າງໆວ່າຂອງກັບການແກ້ປັນຫາໃນຊ່ວງນັ້ນ
 3. ກົນໄກໃນການຄວບຄຸມ ເປັນສ່ວນທີ່ກຳນົດລຳດັບຂອງກົດທີ່ຈະນຳມາໃຊ້ຫຼືປົງທູບກັບຖານຂໍ້ມູນ ເພື່ອບອກລຳດັບຂອງການແກ້ປັນຫາ ສໍາລັບກົນໄກໃນການຄວບຄຸມນີ້ມີຢູ່ 3 ສິ່ງທີ່ສໍາຄັນຕື່:
 - ການກຳນົດທິດທາງສໍາລັບການຄົ້ນຫາ
 - ຂະບວນການໃນການເລືອກກົດເກມ ແລະ
 - ການຄົ້ນຫາແບບຮົວໂລກສະຕິກ

2.1.4 วิธีการต่างๆ ในงานแก้ปัมพา

ເຊິ່ງເປັນເລື່ອງຍາກຫຼາຍທີ່ຈະອາໄສການຄົ້ນຫາດໍາຕອບດ້ວຍວິທີການຂອງການຄົ້ນຫາຂໍ້ມູນທຳມະດາ ໃນການແກ້ປັນຫາຫາຂອງປັນຍາປະດິດນັ້ນ ມີເລື່ອງໃຫຍ່ທີ່ຈະຕ້ອງພິຈາລະນາເຖິງຄື:

ຈະໄດ້ກ່າວຢ່າງລະອຽດໃນຂໍ້ຕໍ່ໄປ. ສຳລັບໂຄງສ້າງຂໍ້ມູນ ໂດຍສ່ວນໃຫຍ່ແລ້ວຈະຖືກກຳນົດໃຫ້ຢູ່ໃນຮູບຂອງໂຄງສ້າງຕົນໄມ້ ຫຼື ໂຄງສ້າງກາບ ເຊິ່ງໂຄງສ້າງທັງສອງນີ້ສາມາດປັບປຸງແບ່ງໄປມາໄດ້ ຕົວຢ່າງໆ ຂອງໂຄງສ້າງຂໍ້ມູນແບບຕົນໄມ້ ແລະກາບໄດ້ສະແດງໄວ້ດ້ວຍ ຮູບທີ 3 ເຊິ່ງເປັນເລື່ອງຂອງໄຕມີ້າ. ການສ້າງຮູບແບບເຄື່ອຄ່າຍຂໍ້ມູນແບບກາບໄດ້ຍົກວ່າໄປແລ້ວຈະດີກວ່າແບບຕົນໄມ້ ເນື່ອງຈາກຈະເຮັດໃຫ້ປະຍັດເນື້ອທີ່ຂອງໜ່ວຍຄວາມຈຳແລະການຄົ້ນຫາຂໍ້ມູນເຮັດໄດ້ໄວກວ່າ ແຕ່ຂໍເສຍກຳລິທຳການເຊື່ອມຕໍ່ຂອງຂໍ້ມູນເຮັດໄດ້ຍາກແລະໃຊ້ເວລາໃນການເຊື່ອມຕໍ່ຂໍ້ມູນດົນກວ່າ ດັ່ງນັ້ນໃນການອອກແບບໂຄງສ້າງບາງເທື່ອຈຶ່ງນິຍົມສ້າງເປັນແບບຕົນໄມ້ກ່ອນແລ້ວຍຈຶ່ງປົງນເປັນກາບການປ່ຽນຂະບວນການຄົ້ນຫາໃນໂຄງສ້າງຕົນໄມ້ (Tree Search Procedure) ເປັນຂະບວນການຄົ້ນຫາໃນໂຄງສ້າງກາບ (Graph Search Procedure)ສາມາດເຮັດໄດ້ໂດຍວິທີການດັ່ງຕໍ່ໄປນີ້:

1. ກວດສອບກຸ່ມຂອງໂນດທີ່ໄດ້ສ້າງມາແລ້ວວ່າ ໂນດທີ່ຈະສ້າງໃໝ່ມາແລ້ວຫຼືບໍ່
2. ຖ້າຍັງບໍ່ມີ ກໍ່ຈະສ້າງໂນດໃໝ່ໃນລະດັບຖຸດໄປໃຫ້ກັບໂຄງສ້າງຕົນໄມ້
3. ຖ້າມີ ໃຫ້ກະທຳດັ່ງຕໍ່ໄປນີ້
 - ຕັ້ງໃຫ້ໂນດທີ່ຈະສ້າງໃໝ່ຊື້ໄປທີ່ໂນດທີ່ມີຢູ່ແລ້ວ ແລະເອົາໂນດທີ່ຈະເກີດຖື້ມໄປ

2.1.5 ການສະແດງຄວາມຮູ້

ເປັນຂະບວນການໃນການສະແດງຄວາມໝາຍທີ່ປາກີດຢູ່ໃນແຕ່ລະໂນດວ່າຈະມີ ວິທີການແນວໃດ, ສຳລັບເລື່ອງກ່ຽວກັບການສະແດງຄວາມຮູ້ຈະໄດ້ມີການອະທິບາຍລະອຽດໃນບົດຕໍ່ໄປ. ໃນກໍລະນີຂອງໄຕມີ້າດັ່ງທີ່ໄດ້ກ່າວໄວ້ໃນບົດນີ້ ອາໄສວິທີການສະແດງຄວາມໝາຍໄດ້ໃຊ້ຊຸດຂອງຕົວເລກໄດ້ຕົວທຳອິດສະແດງການສະແດງຄວາມຮູ້ນອກຈາກການກຳນົດວິທີການທີ່ຈະສະແດງຄ່າຂອງໂນດແລ້ວການສະແດງຄວາມຮູ້ຍັງຈະຕ້ອງປະກອບດ້ວຍສ່ວນທີ່ສະແດງຄວາມສຳພັນຂອງເງື່ອນໄຂການປ່ຽນແບ່ງສະຖານະດັ່ງນີ້:

$$(X < 4, Y) \rightarrow (4, Y)$$

ສິນເຜີນຂ້າງຕົ້ນເປັນການສະແດງຄວາມສຳພັນຂອງການປ່ຽນສະຖານະ ເຊິ່ງອະທິບາຍໄດ້ວ່າ ຖ້າ $(X < 4, Y)$ ຄື ຄ່າຂອງ X ນ້ອຍກວ່າ 4 ແລະ ຄ່າຂອງ Y ເປັນຄ່າໃດ ແລ້ວ ເປັນຈຶ່ງ ເຮົາສາມາດທຳໃຫ້ຄ່າຂອງ X ເປັນ 4 ແລະ ຄ່າ Y ຄົງທີ່ໄດ້.

ວິທີດັ່ງກ່າວເປັນການສະແດງຄວາມຮູ້ແບບງ່າຍໆ ແຕ່ໃນກໍລະນີທີ່ຄວາມຮູ້ມີຄວາມຊັບຊັອນຂຶ້ນມີເລື່ອງໝາຍຢ່າງທີ່ຈະຕ້ອງພິຈາລະນາ ເຊັ່ນ: ໃນກໍລະນີທີ່ຄວາມຮູ້ບໍ່ແມ່ນຕົວເລກ ແຕ່ແມ່ນອອບເຈັກ (object) ແລະ ຄວາມຈຶ່ງ (fact) ທີ່ມີຄວາມສຳພັນກັນ ເຊັ່ນ: ຄວາມຮູ້ 'Plant is on the table' ຈະມີອອບເຈັກ 2 ຕົວທີ່ Plant ແລະ Table ທີ່ມີ on ສະແດງເຖິງຄວາມສຳພັນ ການສະແດງຄວາມຮູ້ແບບນີ້ຈະມີການກ່າວເຖິງຕໍ່ໄປຢ່າງລະອຽດ ສຳລັບຕົວຢ່າງງຂອງການສະແດງຄວາມຮູ້ແບບນີ້ມີຕົວຢ່າງດັ່ງຕໍ່ໄປນີ້:

ON(plant,table) : plant is on the table

IN(table,room) : table is in the room

UNDER(table,window) : table is under the window

ຢ່າງໄດ້ຕາມການສະແດງຄວາມຮູ້ມີສິ່ງທີ່ຄວນຈະຄໍານິງເຖິງດັ່ງຕໍ່ໄປນີ້:

1. ຄວາມຮູ້ທັງໝົດຈະສາມາດລວມເປັນຄວາມຮູ້ດັງກຳໄດ້ແນວໃດ ເຊັ່ນຫາກວ່າເຮົາກຳລັງອະທິບາຍເຖິງລັກສະນະຂອງຫ້ອງຫ້ອງໜຶ່ງທີ່ບອກວ່າ ຫ້ອງນີ້ຕັ້ງ ໂຕະໄວ້ກ້ອງບ້ອງຢັ້ງ 'table is under the window' ແລ້ວມີໜຶ່ງເມື່ອມີການປ່ຽນຖານຄວາມຮູ້ວ່າ CENTER(table,room) ໃນລະບົບການສະແດງຄວາມຮູ້ຈະມີວິທີການແນວໃດທີ່ຈະ

ເຮັດໃຫ້ຮູ້ວ່າ UNDER(table,window) ໃຊ້ບໍ່ໄດ້ແລ້ວເພາະເມື່ອໄຕມາຢູ່ກາງຫ້ອງກຳເປັນໄປບໍ່ໄດ້ທີ່ໄຕ ໂຕະໂຕດງວັນຈະຢູ່ກ້ອງບ້ອງຢັ້ງ.

2. ການຈັດລຳດັບແນວໃດເຮັດໃຫ້ການຄົ້ນຫາເຮັດໄດ້ງ່າຍ ເຊັ່ນ ຖ້າຈະເຕີມ ABOVE(ceiling, floor) ເຂົ້າໄປໃນຖານຄວາມຮູ້ ຈະໃສ່ບ່ອນໃດທີ່ຈະໄດ້ບໍ່ຕ້ອງບອກທຸກເທື່ອເມື່ອມີການກ່າວອ້າງເຖິງເລື່ອງຂອງຫ້ອງເພາະ ' ເພດານຢູ່ເໜືອພື້ນ' ນີ້ເປັນຄວາມຈີງທີ່ວ່າໄປໃນເລື່ອງທີ່ກ່ຽວກັບຫ້ອງ.

ຂະບວນການດັ່ງກ່າວມາຂ້າງເທິງຫ້ອງໝົດ, ຄວາມຈີງແລ້ວຈະເປັນເລື່ອງທີ່ກ່ຽວກັບການສະແດງຄວາມຮູ້ ໂດຍເຟົມ(frame) ເຊິ່ງຈະໄດ້ມີການອະທິບາຍລະອຽດ ໃນບົດທີ 3

2.1.6 ຂະບວນການໃນການເລືອກກົດເການ

ໂດຍປຶກກະຕິແລ້ວວິທີໃນການເລືອກກົດເການສາມາດເຮັດໄດ້ໂດຍວິທີການປົງບໜັບ(matching) ເຊິ່ງເຮັດໄດ້ໂດຍການນຳສະຖານະປັດຈຸບັນ(current state) ໄປປົງບໜັບກັບ ເງື່ອນໄຂຂອງກົດ ຂະບວນການຂອງການປົງບໜັບທີ່ນີ້ມີໃຊ້ກັນໜ້າຍມີຢູ່ 2 ວິທີຕື່

1. ການເຮັດດັດນີ້ (Indexing)

ໜັກການຂອງການແກ້ປັນຫາແບບນີ້ຕີໃຫ້ຫາກົດທຸກຂີ້ທີ່ເງື່ອນໄຂກົງກັບເງື່ອນໄຂ ຫຼືວ່າງເອົາໄວ ໃນສາມານະປັດຈຸບັນແລ້ວດີງເອົາກົດທຸກຂັ້ນນັ້ນອອກມາ ເພື່ອຈະທຳການປົງບໜັບ ຖ້າຫາກວ່າໄດ້ຄໍາຕອບການແກ້ປັນຫາກໍສິ້ນສຸດ ຖ້າຫາກວ່າບໍ່ພົບຄໍາຕອບ ໃຫ້ທຳການປົງບໜັບໃໝ່ຈົນພົບຄໍາຕອບ.

ຖ້າເບິ່ງຈາກເລື່ອງຂອງໄຕນັ້ນເຮົາສາມາດອະທິບາຍກົດໃນລັກສະນະຂອງເງື່ອນໄຂແລະຂໍສະຫຼຸບໄດ້ດັ່ງຮູບທີ່ 2.4 ໃນສ່ວນຂອງເງື່ອນໄຂເປັນສ່ວນທີ່ບອກເງື່ອນໄຂຂອງການໃຫ້ກົດ ແລະ ໃນສ່ວນຂອງການສ້າງເປັນສ່ວນທີ່ບອກເຖິງສະຖານະໃໝ່ທີ່ຈະເປັນຂອງສະຖານະຕົວຢ່າງ ເບິ່ງທີ່ກົດຂີ້ທີ່ 1 ຈາກຕາຕະລາງ ໃນຮູບທີ່ 2.9 ຈະສາມາດເຊົ້າໃຈໄດ້ວ່າຖ້າຫາກໃນສະຖານະປັດຈຸບັນ 'ໄຕນັ້ນທີ່ 1 ມີນັ້ນອຍກວ່າ 4 ລິດ' ແລ້ວໃຫ້ 'ເຕີມນັ້ນລົງໃນໄຕນັ້ນທີ່ 1 ຈົນເຕັມ' ຜົນທີ່ໄດ້ຈະເປັນ 'ໄຕນັ້ນທຳມືດນັ້ນທີ່ 4 ລິດ ແລະ ໄກທີ່ສອງມີນັ້ນຢູ່ເກົ່າເຕີມ(4,Y)'.

ຖ້າຫາກຈະພື້ຈາລະນາຈາກສະຖານະເລີ່ມຕົ້ນທີ່ມີນັ້ນໃນໄຕນັ້ນທັງສອງເປັນ 0 ຊື້ (0,0) ແລ້ວ ພົຈາລະນາຈາກເງື່ອນໄຂຕາມຕາຕະລາງໃນຮູບທີ່ 2.4 ຈະເຫັນໄດ້ວ່າ ມີກົດຂີ້ທີ່ 1 ແລະ 2 ເກົ່ານັ້ນທີ່ມີເງື່ອນໄຂທີ່ກົງກັນ.

ເມື່ອນຳເອົາກົດຂີ້ທີ່ 1 ມາໃຊ້ ສະຖານະຈະປ່ຽນຈາກ(0,0) ເປັນ(4,0)

ເມື່ອນຳເອົາກົດຂີ້ທີ່ 2 ມາໃຊ້ ສະຖານະຈະປ່ຽນຈາກ(0,0) ເປັນ(0,3)

ຈາກສະຖານະ (4,0) ມີກົດຂີ້ 2, 5 ແລະ 8 ທີ່ມີເງື່ອນໄຂກົງກັນ ແລະ ສະຖານະຕ່າງໆ ຈະປ່ຽນໄປດັ່ງນີ້

(4,0) -> (4,3) ໂດຍການໃຊ້ກົດຂໍ 2

(4,0) -> (0,0) ໂດຍການໃຊ້ກົດຂໍ 3

(4,0) -> (1,3) ໂດຍການໃຊ້ກົດຂໍ 6

ຈາກສະຖານະ (0,3) ມີກົດຂໍ 1, 6 ແລະ 9 ທີ່ມີເງື່ອນໄຂກົງກັນ ສະຖານະຕ່າງໆ ຈະປ່ຽນໄປເປັນດັ່ງນີ້

(0,3) -> (4,3) ໂດຍກົດຂໍ 1

(0,3) -> (0,0) ໂດຍກົດຂໍ 4

(0,3) -> (3,0) ໂດຍກົດຂໍ 7

ຖ້າເບິ່ງຈາກໂຄງສ້າງຕົນໄມ້ທີ່ສ້າງຂຶ້ນ ຈະເຫັນວ່າມີຢູ່ໜ້າຍສະຖານະທີ່ບໍ່ມີການສ້າງສະຖານະໃໝ່ ເນື່ອງຈາກວ່າສະຖານະເຫຼົ່ານັ້ນໄດ້ເກີດຂຶ້ນມາແລ້ວກ່ອນໜ້ານັ້ນ ໃນລະດັບທີ່ສູງກວ່າ ເປັນແນວນີ້ເພາະຖ້າຫາກວ່ານຳສະຖານະເຫຼົ່ານັ້ນມາສ້າງສະຖານະໃໝ່ ຈະຮັດໃຫ້ເກີດການວິນລຸບ (loop) ຂຶ້ນ ເຊິ່ງກຳໝາຍຄວາມວ່າບໍ່ມີຫາງຈະພົບຄຳຕອບ

ນອກຈາກປັນຫານີ້ແລ້ວ ໃນການແກ້ປັນຫານີ້ມີປັນຫາອື່ນທີ່ຕ້ອງພິຈາລະນາ

- ຕ້ອງໃຊ້ກົດໝາຍຂໍເພື່ອອະນຸຍາຍແຕ່ລະເງື່ອນໄຂ
- ການຄົ້ນຫາໃຊ້ເວລາດິນ

2. ການຈັບຄູ່ກັບຕົວປຸ່ງ (Matching with variable)

ວິທີນີ້ຂຶ້ນຢູ່ກັບການປູບທຸກກົດແລະຄວາມຈີງ ເພື່ອຄວາມເຂົ້າໃຈໃຫ້ເບິ່ງຕົວຢ່າງຕໍ່ໄປນີ້:

ຄວາມເປັນຈີງ: ແດງໃຫຍ່ເປັນພໍ່ຂອງແດງ

ຄວາມເປັນຈີງ: ແດງເປັນພໍ່ຂອງແດງນັຍ

ຈາກຄວາມເປັນຈີງດັ່ງກ່າວ ຖ້າຈະຖານວ່າ 'ແດງໃຫຍ່ເປັນຫຍ້ງກັບແດງນີ້' ຄອມພິວເຕີຈະບອກເຖິງຄວາມສຳພັນນີ້ບໍ່ໄດ້ ສິ່ງທີ່ຄອມພິວເຕີຈະບອກເຖິງຄວາມສຳພັນນີ້ໄດ້ ຈະຕ້ອງມີກົດບາງຢ່າງທີ່ກ່າວເຖິງເລື່ອງດັ່ງກ່າວ ເຊັ່ນ:

rule: ຖ້າ X ເປັນພໍ່ຂອງ Y

ແລະ Y ເປັນພໍ່ຂອງ Z

ດັ່ງນັ້ນ X ເປັນປຸ່ງຂອງ Z

ເມື່ອມີຄວາມຈີງແລະກົດດັ່ງທີ່ໄດ້ສະແດງແລ້ວ, ຄວາມເປັນໄປໄດ້ຂອງການຕອບຄຳຖາມຈະເຫັນແຈ້ງຂຶ້ນໂດຍການແກນຄ່າ X ຕື່ແດງໃຫຍ່ ແລະ Y ຕື່ແດງ ແລະ Z ຕື່ແດງນີ້ອຍ ເຮົາກຳຈະໄດ້ຄວາມສຳພັນຂອງແດງໃຫຍ່ ແລະ ແດງນີ້ອຍວ່າ ຕື່ປຸ່ງຕາມກົດ ເຊິ່ງລັກສະນະແບບນີ້ເອງທີ່ເຮົາເຫັນວ່າ ການຫາຂີ້ສະຫຼຸບ

ການຫາຂໍສະບູບດັ່ງທີ່ກ່າວມາມີກົດພຽງຂໍດັງວເທົ່ານັ້ນ ເຊິ່ງເຮັດໄດ້ງ່າຍແຕ່ໃນກໍລະນີທີ່ມີກົດໝາຍຂຶ້ນ ເຊັ່ນ ຖ້າຫາກເພີ່ມກົດເຂົ້າໄປອີກ 1 ຂໍດັ່ງນີ້

rule 1: ຖ້າ X ເປັນອ້າຍຂອງ Y ແລະ

Y เป็นอย่าง Z

ດំឡើងន័ែម X បែងចាយខេរា Z

เมื่อเป็นแบบนี้ ปั๊บหากำจังเกิดขึ้นตามม Gov ก็จะเลือกภารกิจ genres ขึ้นมาในภารกิจ genres ที่ได้ในภารกิจ genres ของ rule และ rule1 ในนี้หากำจังที่เกิดขึ้นมาจาก fact ที่มีอยู่ จะเห็นได้ว่าจะต้องเลือก rule เพาะ ความสำคัญของ 'ແດງໃຫຍ່' 'ແດງ' และ 'ແດງນົອຍ' เป็น 'ພື້ນມີແມ່ນ' 'ອ້າຍ' 'ດັ່ງນັ້ນໃນທາຂໍສະຫຼຸບສ່ວນທີ່ເຮັດໜ້າ ທີ່ໃນ

ທາຂໍ້ສະຫຼຸບຈະຕົອງກວດສອບຄວາມສໍາພັນ(relation 'ຝໍ' ແລະ 'ອ້າຍ') ຂອງ ແອດທີ່ບິວ (attribute 'ແດງ' ໄທຍ່' 'ແດງ' ແລະ 'ແດງນ້ອຍ') ວ່າກ່ຽວພັນກັບກົດຂໍ້ໄດ ຈຶ່ງນຳກົດຂັ້ນນັ້ນໄປທາຂໍ້ສະຫຼຸບ

วิธีการแก้ปัญหาแบบนี้ จะต้องอาศัยวิธีการสังเคราะห์ความรู้ที่แตกต่างจากการสังเคราะห์ความรู้ในเลืองๆ หนึ่งด้วยที่สังเคราะห์นั่นมา เป็นการแก้ปัญหาของความรู้ที่ทີກสังเคราะห์โดยกีดกันสังเคราะห์ความรู้อีกแบบหนึ่งที่ sama ได้แก่ Predicate Logic

ຕົວຢ່າງຂອງການສະແດງຄວາມຮັດ້ງກ່າວ

Facts:

SON (John,Mary) :John ເປັນລູກຊາຍຂອງ Mary

SON (Bill,John) :Bill ເປັນລູກຊາຍຂອງ John

SON (Tom,Bill) :Tom ເປັນລົກຊາຍຂອງ Bill

SON (Joe,Bill) :Joe ເປັນລູກຊາຍຂອງ Bill

DAUGHTER (Sue,John) :Sue ເປັນລູກສາວຂອງ John

1. SON(x,y) ^ SON(y,z) -> GRANDSON(x,z) :

ຖ້າ x ເປັນລູກຊາຍຂອງ y ແລະ y ເປັນລູກຊາຍຂອງ z
ດັ່ງນັ້ນ x ຈະເປັນຫຼາມຊາຍຂອງ z

2. DAUGHTER(x,y) ^ SON(y,z) -> GRANDAUGHTER (x,z) :

ຖ້າ x ເປັນລວກສາວຂອງ y ແລະ

y เป็นลักษณะ z

ດັ່ງນັ້ນ x ຈະເປັນຫຼາມສາວຂອງ z

3. $\text{SON}(x,y) \wedge \text{DAUGHTER}(y,z) \rightarrow \text{GRANDSON}(x,z)$:

ຖ້າ x ເປັນລູກຊາຍຂອງ y ແລະ

y เป็นลักษณะ z

ດំណឹង x ຈະបែងច្នាមសាហខោរា z

ในภาคเลือกภารกิจ เนื่องจากภาคพิจารณาจ่ายว่า เธอจะมีวิธีการเลือกภารกิจตามข้อใดแล้วก็ตามเลือกภารกิจตามยังจะต้องสืบในภาคกำหนดลำดับของภาคพิจารณาภารกิจโดย ถ้าภาคพิจารณาจ่ายพิจารณาภารกิจข้อใดก่อน ที่ ทั้งเพาะลำดับของภาคพิจารณาภารกิจ จะมีผู้ตัดสินภารกิจปั้นหา โดยปีภารกิจตัดสินภารกิจตามลำดับของภาคพิจารณาคืนหา มี 2 ปีๆ ถ้า

- ການຫາເຫດຜົນແບບໄປໜ້າ(Forward Reasoning) ແລະ
 - ການຫາເຫດຜົນແບບຍົອນກັບໜ້າ(Backward Reasoning)

งานทางเดินแบบไปข้ามและแบบย้อนรักษาไว้เป็นข้อความของภาษาที่มีความหมาย
สำลับกับภาษาถิ่นทาง และภาษาเลือกภูมิภาค เช่นภาษาสิม เพื่อที่จะทำให้ภาษาที่ทางว่าในทางเข้าสังฆะ จะ^{เข้าสังฆะ}
เลื่อนตัวภาษาทางเดินของภาษาส่วนใหญ่ของประเทศปัจจุบัน(Problem Space) และจัดภูมิภาค (ที่ ท่าน^{ภูมิภาค}
ความรู้: Knowledge Base)ให้เหมาะสมกับภาษาถิ่นทางเดินแบบนั้น. โดยปีภูมิภาคตัวแล้วจะอยู่ในส่วน^{ภูมิภาคตัว}
โครงสร้างข้อมูลแบบต้นไม้(Tree Structure) และ / ที่แบบกราฟ (Graph) เป็นรากแบบบัญญา.

ການຫາເທດຜົນແບບໄປໜ້າ ຈະເລີ່ມຕົ້ນຈາກສະຖານະເລີ່ມຕົ້ນນີ້ມາເປັນຮາກ (Root) ຂອງໂຄສ້າງຕົ້ນໄມ້ ແລະສ້າງໂນດລູກ (Successor) ໃນລະດັບຖຸດໄປ ໂດຍການຫາກົດທຸກຂັ້ນ ເຊິ່ງຫາງດ້ານຊ້າຍຂອງກົດທີ່ກົງກັບໄນດຮາກ ແລະໃຊ້ຫາງດ້ານຂວາຂອງກົດສ້າງເປັນລູກຂອງໂນດນັ້ນ ແລະສ້າງລະດັບຕໍ່ໄປໂດຍການເອົາໄນດລູກນັ້ນມາທຳໂດຍວິທີຄູວກັນຈົນຄົບທຸກໄນດຈົນໝີໃນລະດັບນັ້ນ ແລະໃຫ້ເຮັດແບບນີ້ໄປເລື້ອງຈົນຄົບກົດທຸກຂັ້ນ. ໃນການຫາເທດຜົນແບບນີ້ບາງເທື່ອຈະເອີ້ນວ່າ *Data-Driven Reasoning* ເພະການຄົ້ນຫາເລີ່ມຈາກຂໍ້ມູນພາຍໃນກົດ.

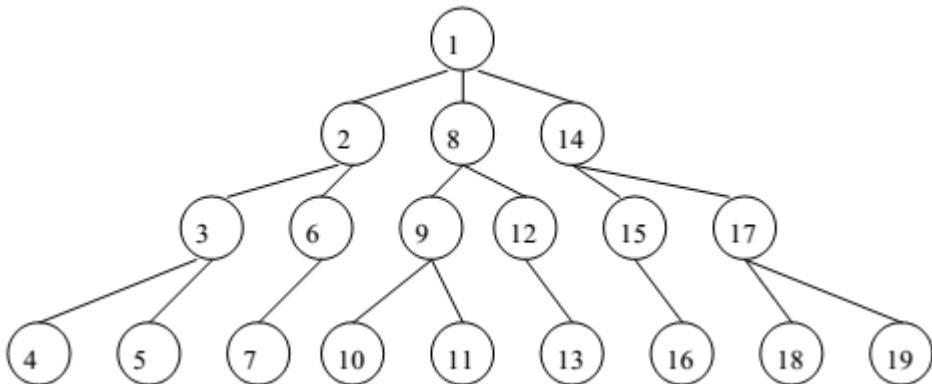
ການຫາເຫດຜົນແບບຍືອນກັບໜັງ ຈະເລີ່ມຕົ້ນຈາກສຖານະເວົ້າໝາຍ ໂດຍການນຳເອົາເວົ້າໝາຍ
ນັ້ນມາສ້າງເປັນຮາກຂອງໂຄງສ້າງຕົ້ນໄມ້ ແລະສ້າງລູກໃນລະດັບທັດໄປ ໂດຍການຫາກົດທຸກຂໍ້ ເຊິ່ງທາງ
ດ້ານຊ້າຍຂອງກົດກົງກັບໂນດຮາກ ແລະໃຊ້ທາງດ້ານຂວາຂອງກົດສ້າງເປັນລູກຂອງໂນດນັ້ນ ແລະສ້າງ
ລະດັບຕໍ່ ໄປໂດຍການເອົາລູກໂນດນັ້ນມາສ້າງວິທີດູວກັນຈົນຄົບທຸກໂນດ ຈົນໝົດໃນລະດັບນັ້ນ ແລະໃຫ້
ຮັດແບບນີ້ໄປເລື້ອຍໆ ຈົນຄົບກົດທຸກຂໍ້ ໃນການຫາເຫດຜົນແບບນີ້ບໍ່ເກີດຈະເຮັ້ນວ່າ: Goal-Driven
Reasoning ເພະການຫາເຫດຜົນເລີ່ມຕົ້ນຈາກເວົ້າໝາຍ (Goal)

2.2. ການຄົ້ນຫາແບບງົມມືດ (Blind Search)

ภารกิจค้นหาแบบบูรณาภิภาค (Blind search) เป็นภารกิจค้นหาแบบที่เดินทางจากโนดหนึ่งไปยังอีกโนดหนึ่ง โดยอาศัยตัวชี้ที่ดินทางเป็นตัวกำหนดภารกิจค้นหา บ์ต้องมีข้อมูลที่บ่งบอกว่าในภารกิจตัดสินใจว่าจะเดินทางต่อไปแบบใด นี่ เว้าอิภารกิจค้นหาแบบที่เดินทางต่อไปจะเลือกเดินทางตามเส้นทางใดมาช่วยในการค้นหาต่อไป บ์ต้องอาศัยเครื่องมือ ของภารกิจที่ดินทาง ตัวอย่างเช่นภารกิจค้นหาข้อมูลแบบนี้คือ ภารกิจค้นหาแบบลึกกว่อง (Depth First Search) และ ภารกิจค้นหาแบบกว้างๆ ก่อน (Breadth First Search)

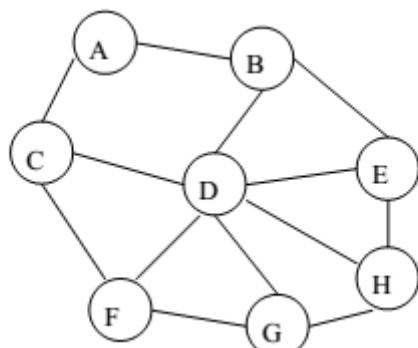
2.2.1 ການຄົ້ນຫາແບບລວງເລີກກ່ອນ

ການຄົ້ນຫາແບບເລີກກ່ອນເປັນການຄົ້ນຫາທີ່ກຳນົດທີ່ມີການຄົ້ນຫາຈາກຮູບຂອງໂຄງສ້າງຕົ້ນໄມ້ ທີ່ເລີ່ມຕົ້ນຈາກໂນດຮາກ (Root node) ທີ່ຢູ່ເທິງສຸດແລ້ວລົງມາໃຫ້ເລີກທີ່ສຸດ ເມື່ອຮອດໂນດລຸ່ມສຸດ(Terminal Node) ໃຫ້ກັບຂຶ້ນມາທີ່ຈຸດສູງສຸດຂອງກົງດັງວກັນທີ່ມີກົງແຍກ ແລະ ຍັງບໍ່ໄດ້ແລ້ວເລີ່ມລົງໄປຈົນຮອດໂນດເລີກສຸດອີກ ເຮັດແບບນີ້ສະຫຼັບໄປເລື້ອຍໆຈົນພົບໂນດທີ່ຕ້ອງການຫາ ຫຼື ກວດສອບຄົບຖຸກໂນດ ແລ້ວຕາມຮູບທີ່ 2.10 ການຄົ້ນຫາແບບເລີກກ່ອນຈະມີລຳດັບການໄປຕາມໂນດຕັ້ງຕົວເລັກທີ່ລະບຸໄວ້ໃນແຕ່ລະໂນດ



ຮູບທີ່ 2.10 ລຳດັບການຄົ້ນຫາແບບລວງເລີກກ່ອນແບບໂຄງສ້າງຕົ້ນໄມ້

ດັ່ງທີ່ໄດ້ກ່າວມາແລ້ວວ່າໂຄງສ້າງຂຶ້ມູນທີ່ໃຊ້ສໍາລັບການຄົ້ນຫາມີສາມາດໃຊ້ກັບໂຄງສ້າງກາບໄດ້ໂດຍອາໄສຫຼັກການດຽວກັນ, ແຕ່ສໍາລັບການເດີນຫາງເທິງກາບນັ້ນຈະບໍ່ມີໂນດເລີກສຸດ, ດັ່ງນັ້ນການເດີນຫາງເທິງກາບຈະຕ້ອງປັບວິທີການເປັນດັ່ງນີ້: ໂດຍເລີ່ມຈາກໂນດເລີ່ມຕົ້ນ ຈາກນັ້ນໃຫ້ນຳໂນດທີ່ຢູ່ຕິດກັບໂນດທີ່ກຳລັງກວດສອບຢູ່ (ທີ່ຍັງບໍ່ໄດ້ສ້າງການກວດສອບ ແລະ ຍັງບໍ່ໄດ້ຢູ່ໃນສະແຕ່ກາມໄສ່ສະແຕ່ກ່າ) ມາເກັບໄວ້ໃນສະແຕ່ກ່າ, ເມື່ອກວດສອບໂນດນັ້ນສໍາເລັດໃຫ້ປ່ອບ (Pop) ຕົວເທິງສຸດຂອງໂນດອອກມາສ້າງການກວດສອບ, ແລ້ວນຳໂນດຂ້າງຄົງທັງໝົດທີ່ຍັງບໍ່ໄດ້ກວດສອບມາຕໍ່ທ້າຍສະແຕ່ກ່າ ແລ້ວປ່ອບຕົວເທິງສຸດອອກມາກວດສອບ. ເຮັດແບບນີ້ເລື້ອຍໆຈົນກວ່າພົບໂນດທີ່ຕ້ອງການ ຫຼື ກວດສອບຄົບຖຸກໂນດ



ຮູບທີ່ 2.11 ໂຄງສ້າງຂຶ້ມູນແບບກາບ

ການກວດສອບຈະເລີ່ມຕົ້ນທີ່ A ແລະ ນຳໂນດຂ້າງຄູງ B ແລະ C ມາເກັບໄວ້ໃນສະແຕ່ກາເມື່ອ ກວດສອບ A ສໍາເລັດປ່ອບຂໍ້ມູນຈາກສະແຕ່ກອກອກມາໄດ້ C ສ້າງການກວດສອບ C ແລະ ນຳໂນດຂ້າງຄູງກັບ C ທີ່ຍັງບໍ່ໄດ້ສ້າງການກວດສອບແລະຍັງບໍ່ໄດ້ຢູ່ໃນສະແຕ່ກາມໄສ່ສະແຕ່ກ D ແລະ F ພຸສ (Push)ໃສ່ສະແຕ່ກດັ່ງນັ້ນໃນສະແຕ່ກຕອນນີ້ມີ B D F ຢູ່ ເມື່ອກວດສອບ C ສໍາເລັດ ປ່ອບ F ອອກມາ ສ້າງການກວດສອບແລ້ວນຳໂນດຂ້າງຄູງທີ່ຍັງບໍ່ໄດ້ກວດສອບແລະຍັງບໍ່ໄດ້ຢູ່ໃນສະແຕ່ກາມໄສ່ສະແຕ່ກ ເຊິ່ງກຳຄື G ດັ່ງນັ້ນຂໍ້ມູນໃນສະແຕ່ກຈະເປັນ B D G ເຮັດແບບນີ້ໄປເລື້ອຍໆ ຖ້າ ຈົນຈົບການເຮັດວຽກກຳຈະໄດ້ລຳດັບການກວດສອບຄື (A C F G H E D B) ຕາມຕາຕະລາງ 1 ດັ່ງຕໍ່ໄປນີ້:

ໂນດທີ່ສໍາຫຼວດ	ສະແຕ່ກ
A	B C
C	B D F
F	B D G
G	B D H
H	B D E
E	B D
D	B
B	

ຕາຕະລາງທີ 2.2 ລຳດັບການຄົ້ນຫາແບບລວງເລີກກ່ອນ

ໃນການຄົ້ນຫາຂໍ້ມູນແບບນີ້ເຖິງ ໂຄງສ້າງຂອງການເບ ມີສິ່ງທີ່ຄວາມສັງເກດຄື ໂນດທີ່ເລີ່ມຕົ້ນການກວດສອບຈະຕ້ອງມີການກຳນົດມາໃຫ້ວ່າໂນດໄດ້ເປັນໂນດເລີ່ມຕົ້ນ ແລະ ຂໍ ສັງເກດອີກຍ່າງໜຶ່ງຄືວິທີການຄົ້ນຫາແບບເລີກກ່ອນທີ່ໃຊ້ສໍາລັບ ໂຄງສ້າງຂໍ້ມູນແບບການເບ ສາມາດໃຊ້ກັບ ໂຄງສ້າງຂໍ້ມູນແບບຕົ້ນໄມ້ໄດ້ດ້ວຍ

- ອານກຳລົດທີ່ມີ ການຄົ້ນຫາຂໍ້ມູນແບບເລີກກ່ອນ

ໃຫ້ສະຖານະ 1 ໝາຍເຖິງໂນດທີ່ຍັງບໍ່ກວດສອບ, ສະຖານະ 2 ໝາຍເຖິງໂນດທີ່ຢູ່ໃນ STACK ແລະ ສະຖານະ 3

ໝາຍເຖິງໂນດທີ່ສ້າງການກວດສອບແລ້ວ

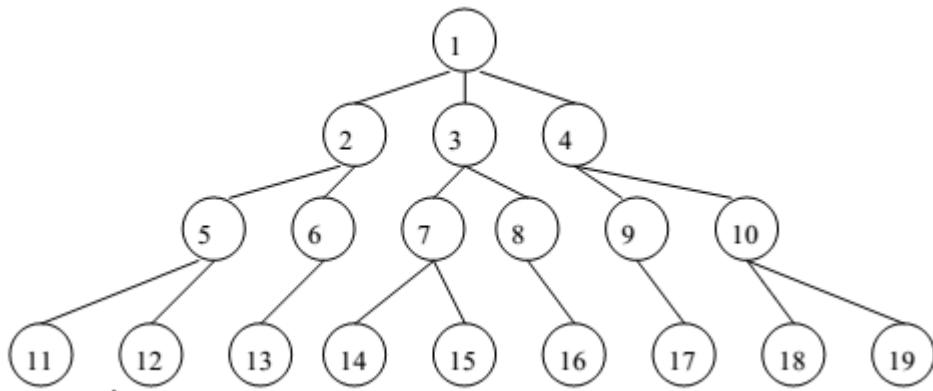
- ທຳໃຫ້ໂນດທຸກໂນດມີສະຖານະເປັນ 1 ແລະ ນຳໂນດເລີ່ມຕົ້ນໄວ້ໃນ STACK ບ່ຽນສະຖານະເປັນ 2
- ນຳໂນດເຖິງສຸດໃນ STACK ອອກມາກວດສອບ ແລະ ບ່ຽນສະຖານະເປັນ 3

- ຖ້າໄນດທີ່ສ້າງການກວດສອບຢູ່ຕື່ໄນດເປົ້າໝາຍລາຍງານໄນດທີ່ກວດສອບຕື່ໄນດເປົ້າໝາຍ ແລະໃຫ້ຂໍາມໄປທີ່ຂັ້ນຕອນ 4
- ຖ້າໄນດທີ່ກວດສອບບໍ່ໃຊ້ໄນດເປົ້າໝາຍ ແລະຈຳນວນໄນດໃນ STACK ມີໝາຍກວ່າ 0 ໃຫ້
ນຳໄນດຂ້າງຄູງທີ່ມີສະຖານະເປັນ 1 ຫ້າງໝົດ(ຖ້າມີ) ປຸ່ງສະຖານະເປັນ 2 ແລ້ວນຳໃສ່ໄວ້ໃນ STACK ແລະ ກັບໄປເຮັດຂຶ້ນທີ່ 2
 - 3) ລາຍງານການຄົ້ນຫາບໍ່ໄດ້
 - 4) ການຮັດວຽກສິນສຸດ

ໝາຍເຫດ ລັກສະນະການໃສ່ຂໍ້ມູນເຂົ້າແລະການນຳຂໍ້ມູນອອກຂອງໂຄງສ້າງຂໍ້ມູນແບບສະແຕ່ກ ຈະເປັນໃນລັກສະນະຂໍ້ມູນທີ່ນຳເຂົ້າກ່ອນ ຈະຖືກນຳມາໃຊ້ໝາຍຫຼັງ

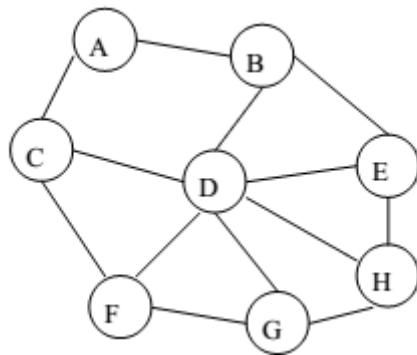
2.2.2. ການຄົ້ນຫາແບບລວງກວ້າງກ່ອນ

ການຄົ້ນຫາແບບລວງກວ້າງກ່ອນເປັນການກຳນົດທີ່ດ້ວຍການຄົ້ນຫາແບບເທື່ອລະລະດັບຂອງໂຄງສ້າງຕົ້ນໄມ້ໂດຍເລີ່ມຈາກໄນດທາກ (ລະດັບທີ່ 0) ແລ້ວລົງມາລະດັບທີ່ 1 ຈາກຊ້າຍໄປຂວາ ເມື່ອສຳເລັດລະດັບທີ່ 1 ໄປລະດັບທີ່ 2 ຈາກຊ້າຍໄປຂວາເຊັ້ນກັນ ເຮັດແບບນີ້ເລື້ອຍໆ ຈິນພົບໄນດທີ່ຕ້ອງການຕາມຮູບທີ່ 3 ລຳດັບການເດີນຫາງຂອງໄນດເປັນໄປຕາມໝາຍເລກທີ່ລະບຸໄວ້ເທິງໄນດ



ຮູບທີ 2.12 ລຳດັບການຄົ້ນຫາແບບລວງກວ້າງກ່ອນແບບໂຄງສ້າງຕົ້ນໄມ້

ສຳລັບການຄົ້ນຫາແບບລວງກວ້າງກ່ອນເທິງໂຄງສ້າງຕົ້ນໄມ້ ຈະອາໄສໂຄງສ້າງຂໍ້ມູນແບບຄົວ (Queue) ມາຊ່ວຍ ແລະ ດ້ວຍວິທີການອັນດັງວກັບການຄົ້ນຫາແບບເລີກກ່ອນຕີ ໃຫ້ເລີ່ມຕົ້ນກວດສອບທີ່ໄນດເລີ່ມຕົ້ນແລ້ວນຳໄນດຂ້າງຄູງເກັບໄວ້ໃນຄົວ ເມື່ອກວດສອບໄນດເລີ່ມຕົ້ນສຳເລັດ ໃຫ້ນຳຂໍ້ມູນໃນຄົວ ອອກມາກວດສອບແລ້ວນຳໄນດຂ້າງຄູງທີ່ຍັງບໍ່ໄດ້ກວດສອບແລະບໍ່ໄດ້ຢູ່ໃນຄົວໃສ່ຄົວໄວ້ ເຮັດແບບນີ້ໄປເລື້ອຍໆ ຈິນພົບໄນດທີ່ຕ້ອງການ ຫຼື ເມື່ອກວດສອບຄົບຖຸກໄນດ.



ຮູບທີ 2.13 ໂຄງສ້າງຂໍ້ມູນແບບການຈຳເປັດ

ການກວດສອບເລີ່ມຕົ້ນທີ່ A ນຳໄນດ້ຂ້າງຄູງ B,Cໄວ້ໃນຄືວເມື່ອກວດສອບ A ສຳເລັດນຳຂໍ້ມູນໃນຄືວຄື B ອອກມາກວດສອບແລ້ວນຳຂໍ້ມູນຂ້າງຄູງຄື D, E ໄສ່ຄືວ ຕອນນີ້ຄືວຈະມີ B D E ຢູ່ແລ້ວນຳ B ອອກມາກວດສອບ. ເຮັດແບບນີ້ເລື້ອຍໆຈະໄດ້ລຳດັບການກວດສອບຂໍ້ມູນຄື (A B C D E F G H)ຕາມຕາຕະລາງທີ່ 2.3

ໂນດທີ່ສໍາຫຼວດ	ຄືວ
A	B C
B	C D E
C	D E F
D	E F G H
E	F G H
F	G H
G	H
H	

ຕາຕະລາງທີ່ 2.3 ລຳດັບການຄົ້ນຫາແບບລວງກວ້າງກ່ອນ

ຄືກັບການຄົ້ນຫາແບບເລີກກ່ອນ ການຄົ້ນຫາແບບລວງກວ້າງກ່ອນໂດຍໃຊ້ໂຄງສ້າງຂໍ້ມູນຄືວມາຊ່ວຍຕົວມີການກຳນົດໂນດເລີ່ມຕົ້ນ ແລະວິທີການນີ້ສາມາດໃຊ້ໄດ້ກັບຂໍ້ມູນເທິງໂຄງສ້າງແບບຕົ້ນໄມ້

- ອານກຳດົດທຶນ ການຄົ້ນຫາຂໍ້ມູນແບບລວງກວ້າງກ່ອນ

ໃຫ້ສະຖານະ 1 ໝາຍເຖິງໂນດທີ່ຍັງບໍ່ກວດສອບ, ສະຖານະ 2 ໝາຍເຖິງໂນດທີ່ຢູ່ໃນ QUEUE ແລະ ສະຖານະ 3 ໝາຍເຖິງໂນດທີ່ສ້າງການກວດສອບແລ້ວ

- 1) ทำให้โนดທຸກໄໂນດມີສະຖານະເປັນ 1 ແລະນຳໄໂນດເລີ່ມຕົ້ນໄວ້ໃນ QUEUE ປັ່ງສະຖານະເປັນ 2
 - 2) ນຳໄໂນດທຳອິດໃນ QUEUE ອອກມາກວດສອບ ແລະປັ່ງສະຖານະເປັນ 3
 - ຖ້າໄໂນດທີ່ສ້າງການກວດສອບຢູ່ຄືໄໂນດເປົ້າໝາຍ ລາຍງານໄໂນດທີ່ກວດສອບຄືໄໂນດເປົ້າໝາຍ ແລະ ໃຫ້ຂ້າມໄປທີ່ຂັ້ນຕອນ 4
 - ຖ້າໄໂນດທີ່ກວດສອບບໍ່ແມ່ນໄໂນດເປົ້າໝາຍ ແລະຈຳນວນໄໂນດໃນ QUEUE ມີຫຼາຍກວ່າ 0 ໃຫ້ນຳໄໂນດຂ້າງຄົງທີ່ມີສະຖານະເປັນ 1 ຫ້າງໝົດ(ຫ້າມີ) ປັ່ງສະຖານະເປັນ 2 ແລ້ວນຳໄສ່ໄວ້ໃນ QUEUE ແລະ ກັບໄປຂໍ້ 2
 - 3) ລາຍງານການຄົ້ນຫາທີ່ບໍ່ໄດ້
 - 4) ການເຮັດວຽກສິນສຸດ

ໝາຍເຫດ: ລັກສະນະການໃສ່ຂໍ້ມູນເຂົ້າແລະການນຳຂໍ້ມູນອອກຂອງໂຄງສ້າງຂໍ້ມູນແບບຄືວ ຈະເປັນໃນລັກສະນະຂໍ້ມູນທີ່ນຳເຂົ້າກ່ອນ ຈະຖືກນຳມາໃຊ້ກ່ອນ

2.3. ການຄົ້ນຫາແບບຮືວຂີສະຕິກາ (Heuristic Search)

ທາງດ້ານປັນຍາປະດິດ ການຄົ້ນຫາຄໍາຕອບອາໄສວິທີການທາງຮູບຮັບສະຕິກ (Heuristic Search) ມີຄວາມແຕກຕ່າງຈາກການຄົ້ນຫາຂໍ້ມູນແບບທຳມະດາຢູ່ທີ່ການຄົ້ນຫາຂໍ້ມູນແບບທຳມະດາເຊື້ອທີ່ສ້າງການ ຄົ້ນຂໍ້ມູນຈະຕ້ອງກວດສອບຂໍ້ມູນເທື່ອລະຕົວ, ບຸກຕົວຈິນຄົບ ແຕ່ຮູບຮັບສະຕິກຈະບໍ່ລົງໄປເຖິງຂໍ້ມູນທຸກຕົວ, ວິທີການນີ້ຈະເລືອກໄດ້ຄໍາຕອບທີ່ເໝາະສົມໃຫ້ກັບການຄົ້ນຫາ ເຊິ່ງມີຂັ້ດີຕື່: ສາມາດສ້າງການຄົ້ນຫາຄໍາຕອບຈາກ ຂໍ້ມູນທີ່ມີຂະໜາດໃຫຍ່ຫຼາຍໆ ໄດ້ ແຕ່ຂໍ້ເສຍຕື່ຄໍາຕອບທີ່ໄດ້ເປັນພຽງຄໍາຕອບທີ່ດີເທົ່ານັ້ນ, ແຕ່ເນື້ອງຈາກວ່າປັນຫາໃນບາງລັກສະນະນັ້ນໃຫຍ່ຫຼາຍ ແລະ ເປັນໄປບໍ່ໄດ້ທີ່ຈະສ້າງການຄົ້ນຫາດ້ວຍວິທີແບບທຳມະດາຂະບວນການຂອງຮູບຮັບສະຕິກຈຶ່ງເປັນສິ້ທີ່ຈໍາເປັນ.

ໃນເລື່ອງຂອງກິວວິສະຕິການນີ້ ນອກຈາກຈະມີການຄົ້ນຫາແບບກິວວິສະຕິກາແລ້ວ ຍັງມີອີກສິ່ງໜຶ່ງທີ່ສໍາຄັນຕີ ກິວວິສະຕິກັງຊັ້ນ(Heuristic Function) ເຊິ່ງໝາຍເຖິງັງຊັ້ນທີ່ທຳຫັນທີ່ໃນການວັດຂະໜາດຂອງຄວາມເຢັນໄປໄດ້ໃນການເກົ່າປັນຫາເຊື່ອຈະສະແດງດ້ວຍຕົວເລກ

ຈຸດປະສົງແທ້ຈິງຂອງຮົວຮີສະຕິກ ພັງຂັນກຳຄື ການລະບຸທີ່ດທາງຂອງຂະບວນການຄົ້ນຫາ ເພື່ອໃຫ້ຢູ່ໃນທີ່ດທາງທີ່ໄດ້ປະໂຫຍດສູງສຸດ ໂດຍການບອກວ່າເຮົາຄວນເລືອກໄປເສັ້ນທາງໄດ້ກ່ອນ ໃນກໍລະນີມີເສັ້ນທາງໆທາຍກວ່າໜຶ່ງເສັ້ນທາງຕົວໆເລືອກ.

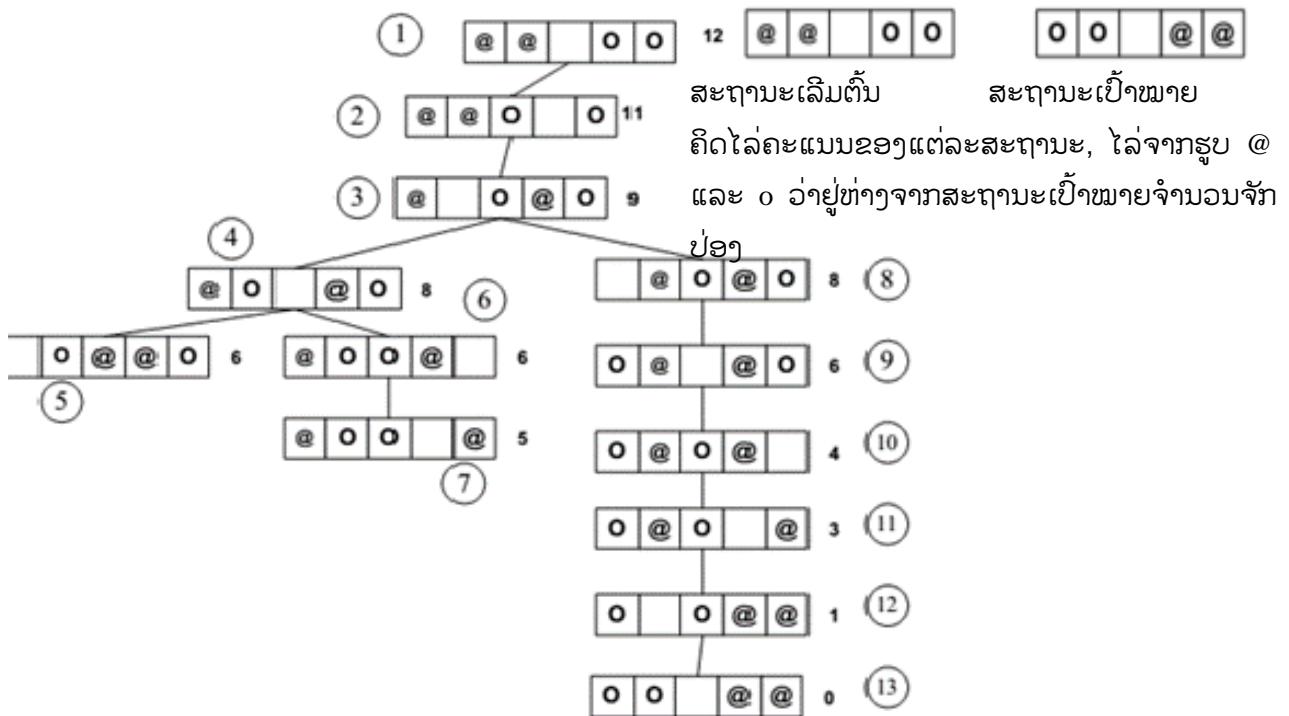
2.3.1. တော်ချေပွဲ (Hill Climbing)

ການຄົ້ນຫາແບບໄຕຂຶ້ນພູ (Hill Climbing) ເປັນວິທີການຄົ້ນຫາຂຶ້ນມູນທີ່ມີລະກະສະນະຄືກັບການໄຕຂຶ້ນພູ ສິ່ງທີ່ນັກໄຕຂຶ້ນພູຈະເດີນທາງໄປຮອດຍອດພູເຊົາ ນັກປິນເຂົາຈະຕົ້ງເບິ່ງກ່ອນວ່າຍອດເຂົາຢູ່ທີ່ໄດ້ແລ້ວນັກປິນເຂົາຈະຕົ້ງພະຍາຍາມໄປຈຸດນັ້ນໃຫ້ໄດ້ ລອງນິກພາບຂອງການໄຕຂຶ້ນພູທີ່ໄລ້ນີ້ເບິ່ງເຫັນແຕ່ຍອດ ແລະ ນັກປິນເຂົາກຳລັງໄຕຂຶ້ນພູຢູ່ທາງລຸ່ມທີ່ມີເສັ້ນທາງເຕັມໄປໝົດ ເພື່ອທີ່ຈະເດີນທາງໄປຮອດຍອດພູເຊົາໂດຍໄວ້ທີ່ສຸດນັກປິນເຂົາຈະເບິ່ງໄປທີ່ຍອດເຂົາແລ້ວສັງເກດວ່າທີ່ດີທາງ ໄດ້ທີ່ເມື່ອປິນແລ້ວຈະເຂົາໄກ້ຍອດເຂົາ ແລະຫຼືກລັງທີ່ດີທາງທີ່ໄປແລ້ວຈະເຮັດໃຫ້ຕົນເອງທ່າງຈາກຍອດເຂົາ ນັກປິນເຂົາຈະຕົ້ງເຮັດແບບນີ້ໄປເລື້ອຍໆ ຈົນກວ່າຮອດຍອດເຂົາ.

ຄວາມຈິງແລ້ວການຄົ້ນຫາວິທີນີ້ເປັນວິທີທີ່ດັດແປງມາຈາກ Generate-and-Test ຕ່າງກັນທີ່ວິທີການຂອງ Generate-and-Test ໃຫ້ຄໍາຕອບຂອງຟັງຊັນກວດສອບ (Test Function) ອອກມາເປັນ yes ແລະ no ແຕ່ຂອງ Hill Climbing ຈະໃຫ້ຄໍາອອກມາເປັນວ່າບໍ່ອນໄດ້ໄກ້ທີ່ສຸດ, ເຊິ່ງປະກອບດ້ວຍຂັ້ນຕອນຕ່າງໆ ດັ່ງຕໍ່ໄປນີ້:

2. Simple Hill Climbing

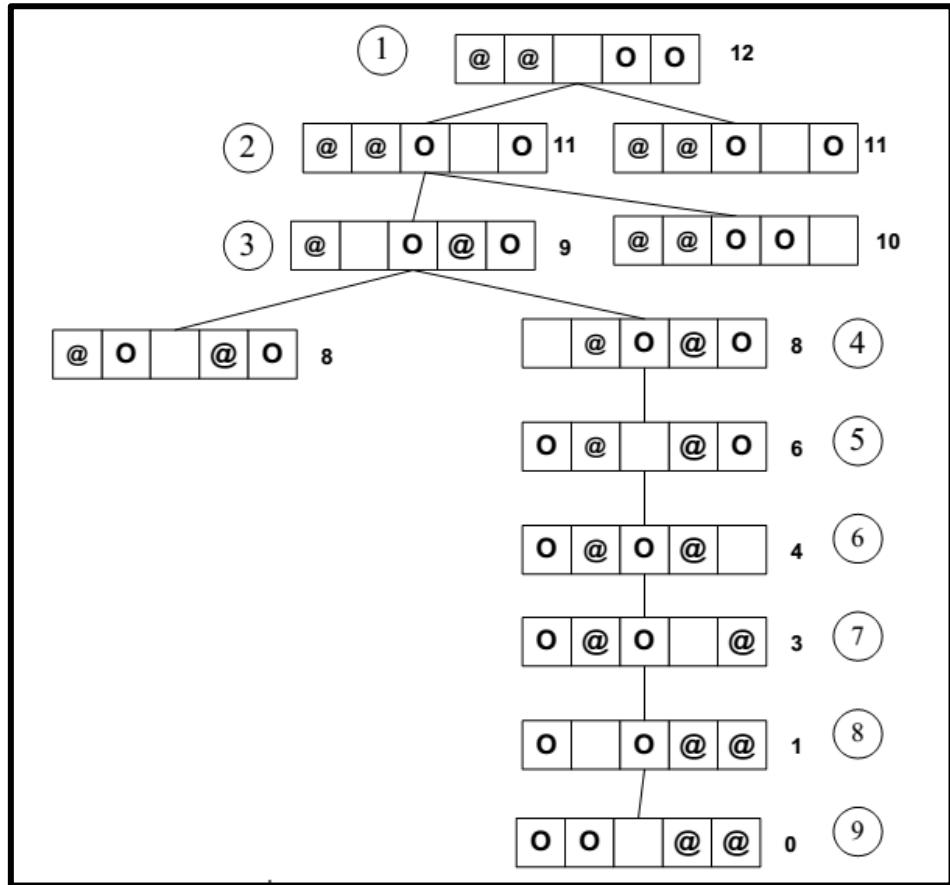
- 1) ກວດສອບສະຖານະເລີ່ມຕົ້ນ ຖ້າສະຖານະເລີ່ມຕົ້ນນີ້ຄືສະຖານະເປົ້າໝາຍ ກໍໃຫ້ສະແດງຄໍາຕອບອອກມາ ແລະຍົກເລີກການເຮັດວຽກ, ແຕ່ຖ້າສະຖານະເລີ່ມຕົ້ນບໍ່ໄດ້ເປັນສະຖານະເປົ້າໝາຍ ກໍປ່ຽນສະຖານະນີ້ເປັນສະຖານະປັດຈຸບັນ (Current State) ແລະ ໄປຕໍ່ຂໍ້ 2
 - 2) ໃຫ້ເຮັດຕາມຂະບວນການທາງລຸ່ມນີ້ຈິນກວ່າຈະພົບຄໍາຕອບ ຫຼື ຈິນກວ່າບໍ່ມີຕົວດຳເນີນການໄດ້ງ໌ທີ່ຈະໃຊ້ກັບສະຖານະປັດຈຸບັນ ເພື່ອສ້າງສະຖານະໃໝ່
 - 2.1) ເລືອກຕົວດຳເນີນການທີ່ຍັງບໍ່ໄດ້ນຳມາໃຊ້ກັບສະຖານະປັດຈຸບັນ ມາໃຊ້ກັບສະຖານະປັດຈຸບັນເພື່ອສ້າງສະຖານະໃໝ່ຂຶ້ນມາສະຖານະໜຶ່ງ
 - 2.2) ກວດສອບສະຖານະໃໝ່ດັ່ງນີ້:
 - ຖ້າເປັນຄໍາຕອບໃຫ້ເລີກການເຮັດວຽກ
 - ຖ້າບໍ່ແມ່ນ ແຕ່ວ່າສະຖານະທີ່ໄດ້ໃໝ່ດີກວ່າສະຖານະປັດຈຸບັນ ໃຫ້ສະຖານະໃໝ່ນີ້ເປັນສະຖານະປັດຈຸບັນ
 - ຖ້າບໍ່ແມ່ນ ແລະ ສະຖານະທີ່ໄດ້ບໍ່ດີເຫັນສະຖານະປັດຈຸບັນ ໃຫ້ກັບເຮັດຊ້າທີ່ຂໍ້ 2 ໃໝ່



ຮູບທີ 2.14 ການແກ້ປັນຫາດ້ວຍ Hill Climbing

2. ໄຕ່ຂຶ້ນຕາມທາງຊັ້ນທີສຸດ (Steepest-Ascent Hill Climbing)

- 1) ກວດສອບສະຖານະເລີ່ມຕົ້ນ ຖ້າສະຖານະເລີ່ມຕົ້ນນີ້ຄືສະຖານະເປົ້າໝາຍ ກໍໃຫ້ສະແດງຄຳຕອບອອກມາ ແລະຍົກເລີກການເຮັດວຽກ, ແຕ່ຖ້າສະຖານະເລີ່ມຕົ້ນບໍ່ໄດ້ເປັນສະຖານະເປົ້າໝາຍ ກໍປ່ຽນສະຖານະນີ້ເປັນສະຖານະປັດຈຸບັນ (current state) ແລະເຮັດຕໍ່ທີ່ຂໍ້ 2
- 2) ໃຫ້ກຳຕາມຂະບວນການຂ້າງລຸ່ມນີ້ຈິນກວ່າຈະພົບຄຳຕອບ ຫຼື ຈິນກວ່າຄາດວ່າບໍ່ສາມາດຫາຄຳຕອບໄດ້
 - 2.1) ໃຫ້ SUCC ເປັນສະຖານະທີ່ບໍ່ວ່າສະຖານະລູກໄດ້ ແລະ ທີ່ເກີດໃໝ່ຈະຕັອງດີກວ່າສະຖານະນີ້
 - 2.2) ສໍາລັບຕົວດໍາເນີນການທີ່ຈະໃຊ້ກັບສະຖານະປັດຈຸບັນແຕ່ລະຕົວ ໃຫ້ດໍາເນີນການດັ່ງນີ້
 - ສ້າງສະຖານະໃໝ່ຈາກສະຖານະປັດຈຸບັນ
 - ກວດສອບສະຖານະໃໝ່ ຖ້າເປັນຄຳຕອບ ໃຫ້ເລີກການເຮັດວຽກ, ຖ້າບໍ່ແມ່ນໃຫ້ປຸງທຸງບສະຖານະໃໝ່ກັບ SUCC ຖ້າສະຖານະທີ່ໄດ້ໃໝ່ດີກວ່າສະຖານະປັດຈຸບັນ ໃຫ້ສະຖານະໃໝ່ນີ້ເປັນ SUCC ຖ້າບໍ່ດີກວ່າ ບໍ່ຕ້ອງເຮັດຫຍັງໝົດ
 - 2.3) ຖ້າ SUCC ດີກວ່າສະຖານະປັດຈຸບັນ ໃຫ້ສະຖານະປັດຈຸບັນຄື SUCC



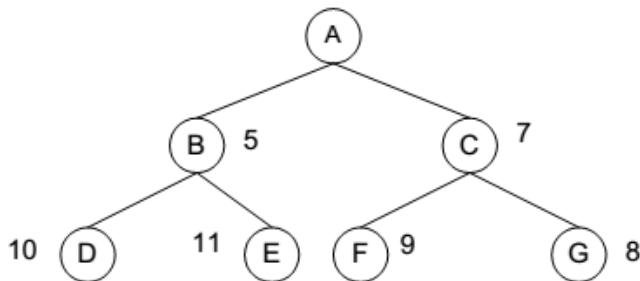
ຮູບທີ 2.15 ສະແດງການແກ້ປັນຫາແບບໄຕ້ຂຶ້ນຕາມທາງຊັ້ນທີ່ສຸດ

ຄວາມແຕກຕ່າງລະຫວ່າງ Hill climbing ທຳມະດາ ກັບແບບ Steepest-Ascent ຄືແບບທຳມະດາຈະເລືອກສະຖານະປັດຈຸບັນຈາກໂນດລູກທີ່ດີກວ່າສະຖານະປັດຈຸບັນຕົວທຳກິດທີ່ພົບໄດຍໍບໍ່ສິນໃຈລູກທີ່ເງື່ອ ແລ້ວສ້າງໂນດຂອງລຸ້ນທັດໄປເພື່ອປູງທູງບຕໍ່ໄປອີກ ແຕ່ສໍາລັບ steepest ascent ນັ້ນຈະເລືອກສະຖານະປັດຈຸບັນຈາກຕົວທີ່ດີທີ່ສຸດຂອງລູກໝົດຊຸດ

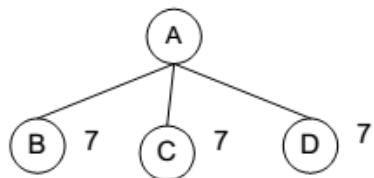
ການແກ້ປັນຫາດ້ວຍ Hill climbing ນັ້ນ ເຖິງວ່າຈະມີຂຶ້ນທູາຍຢ່າງຄ້າຍຄື Generate and Test ແລ້ວ ໃນໂຕຂອງມັນເອງຍັງມີປັນຫາທີ່ອາດຈະເກີດຂຶ້ນໃນການໃຊ້ວິທີການນີ້ຄື:

- Local Maximum: ເປັນສ່ວນທີ່ບອກສະຖານະທີ່ດີທີ່ສຸດເມື່ອທູງບກັບສະຖານະຂ້າງຄ່າງເທົ່ານັ້ນ ແຕ່ທ້າທູງບກັບສະຖານະອື່ນໆ ຫຼືຢູ່ທ່າງອອກໄປ ຫຼື ການກວດສອບຂັ້ນຕໍ່ໄປແລ້ວເປັນໄປໄດ້ວ່າ ຈະໄດ້ຜົນອອກມາດີທີ່ສຸດ ຈາກຮູບ (a) ເຮົາຈະເຫັນວ່າທີ່ໂນດໃນລະດັບທີ 1 ໂນດ B ແລະ C ຄ່າຂອງ B ນ້ອຍກວ່າ C ແຕ່ວ່າລູກຂອງ B ກັບມີຄ່າຫຼາຍກວ່າລູກຂອງ C ດັ່ງນັ້ນໃນການເລືອກໂນດທີ່ທີ່ສຸດໃນລະດັບທີ 1 ຄື C ຈະມີຜົນໃຫ້ໂນດ E ເຊິ່ງຄວນຈະເປັນໂນດທີ່ດີທີ່ສຸດ ບໍ່ໄດ້ຮັບການພິຈາລະນາ ເພາະໃນລະດັບເທິງ ຫຼື ລະດັບທີ 1 ໂນດ B ບໍ່ໄດ້ຮັບການເລືອກ
- ພູພຽງ (Plateau): ໃນກໍລະນີປັນຫາຢູ່ເທິງລະດັບດູວກັນ ແລະ ຜົນຂອງການຫາຄ່າຈາກຮົກວິທີສະຕິກາ

ຟັງຊັ້ນທີ່ໄດ້ເຫົາກັນໝົດ ຈະບໍ່ສາມາດຕັດສິນໃຈໄດ້ວ່າຈະເລືອກເສັ້ນທາງເສັ້ນໄດ້ ຈາກຮູບ (b) ຈະເຫັນວ່າໂນດລູກທັງ 3 ຂອງ A ມີຄ່າເຫົາກັນໝົດເຮັດໃຫ້ເຄີຍບໍ່ສາມາດເລືອກໂນດທີ່ດີທີ່ສຸດໄດ້



(a) ການຄົ້ນຫາທີ່ເກີດບັນຫາແບບ Local Maximum

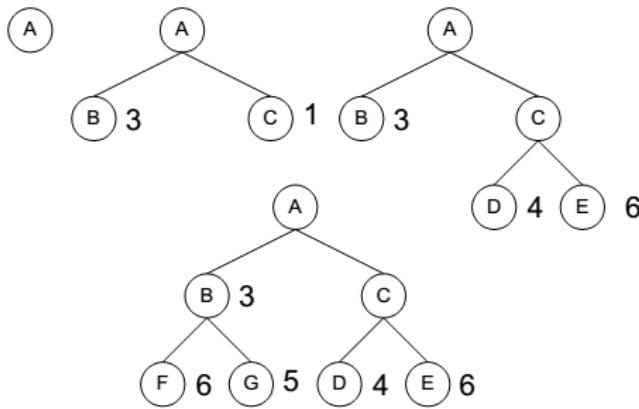


(b) ភាគមិនខាងក្រោមពីរដូចបំបាត់ខាងក្រោម Plateau

- **ສັນເຊົາ** (Ridge): ເປັນກໍລະນີການຄົ້ນຫາໄດ້ໄປໃນທາງທີ່ດີທີ່ສຸດຕະຫຼອດເວລາ ເມື່ອປົງບໜູງກັບໂນດຂ້າງຄູງ ແຕ່ເມື່ອທູນບໜູງກັບໂນດທີ່ດີທີ່ສຸດໃນລະດັບທີ່ແລ້ວຈະບໍ່ດີ ການຄົ້ນຫາໃນລັກສະນະນີ້ຈະມີລັກສະນະແບບດຽວກັບການໄປເທິງສັນເຊົານັ້ນຍ່າງ ທີ່ເຢື່ງຄືວ່າຈະພາໄປສູ່ອດເຊົາໄດ້.

2.3.2. ການຄົ້ນຫາແບບດີທີ່ສດຖອນ(Best-First Search)

ເປັນຂະບວນການຄົ້ນຫາຂໍ້ມູນທີ່ໄດ້ນຳເອົາຂໍ້ດີຂອງທັງການຄົ້ນຫາແບບເລີກກ່ອນ (Depth First Search) ແລະ ການຄົ້ນຫາແບບລວງກວ້າງກ່ອນ (Breadth First Search) ມາລວມກັນເປັນວິທີການດູວໂດຍທີ່ແຕ່ລະຂັ້ນຂອງການຄົ້ນຫາໃນໄນດລູກນັ້ນ ການຄົ້ນຫາແບບດີທີ່ດີກ່ອນຈະເລືອກເອົາ ໂນດທີ່ດີທີ່ສຸດ (Most Promising) ແລະ ການທີ່ຈະຮູ້ວ່າ ໂນດໄດ້ດີທີ່ສຸດນີ້ສາມາດເຮັດໄດ້ໂດຍອາໄສຮິວຮີສະຕິກັບຊັ້ນ ເຊິ່ງຮິວຮີສະຕິກັບຊັ້ນ ນີ້ຈະເຮັດໜ້າທີ່ຄືຕົວວັດຜົນ ແລະ ໃຫ້ຜົນຂອງການວັດນີ້ອອກມາເປັນຄະແນນ ຮູບທີ່ 2.7 ເປັນຕົວຢ່າງຂອງການຄົ້ນຫາແບບດີທີ່ສຸດກ່ອນ ຂັ້ນຕອນນີ້ເລີ່ມຈາກຕອນ 1 ສ້າງໄນດຮາກ (Root Node) ໃນຂັ້ນຕອນ 2 ສ້າງໄນດລູກ B ແລະ C ແລ້ວກວດສອບໄນດ B ແລະ C ດ້ວຍຮິວຮີສະຕິກັບຊັ້ນ ໄດ້ຜົນອອກມາເປັນຄະແນນຄື 3 ແລະ 1 ຕາມລຳດັບ ຈາກນັ້ນໃຫ້ເລືອກໄນດ C ເປັນໄນດຕໍ່ໄປທີ່ເຮົາສິນໃຈ ເພາະມີຄ່າໜ້ອຍກວ່າ (ໝາຍເຫດ ໃນການເລືອກມີຈະເລືອກຄ່າໜ້າຍສຸດ ຫຼືໜ້ອຍສຸດກໍໄດ້ ຂຶ້ນຢູ່ກັບລັກສະນະຂອງປັ້ນຫາ) ແລ້ວສ້າງໄນດ ລູກໃຫ້ກັບໄນດ C ໃນຂັ້ນຕອນ 3 ໄດ້ໄນດ D ແລະ E ແລ້ວກວດສອບຄະແນນໄດ້ 4 ແລະ 6 ຕາມລຳດັບ ຈາກນັ້ນສ້າງການ ປຸງບໜູງບຄ່າຂອງໄນດຫ້າຍສຸດ ຫຼື Terminal Node ທຸກໄນດ ວ່າໄນດ ໄດ້ມີຄ່າດີທີ່ສຸດ ໃນທີ່ນີ້ຈະຕ້ອງເລືອກໄນດ B ເພາະມີຄະແນນພຽງ 3 (ເລືອກຄະແນນຕໍ່ສຸດ) ແລ້ວສ້າງໄນດ ລູກຕາມຂັ້ນຕອນ 4 ໄດ້ F ແລະ G ແລ້ວກວດສອບຄະແນນໄດ້ 6 ແລະ 5 ຄະແນນຕາມລຳດັບ ເຮັດແບບນີ້ເລື້ອຍໆຈົນພິບຄໍາຕອບ ຫຼື ຈົນບໍ່ສາມາດ ສ້າງໄນດຕໍ່ໄປໄດ້ອີກ.



ຮູບທີ 2.16 ການສະແດງຂັ້ນຕອນຂອງການຄົ້ນຫາແບບດີທີ່ສຸດກ່ອນ

- ອານກຳລິດທຶນ: ການຄົ້ນຫາແບບດີທີ່ສຸດກ່ອນ

- ເລີ່ມດ້ວຍ OPEN ທີ່ມີພຽງໄນດ້ເລີ່ມຕົ້ນ
- ຈົນກວ່າຈະພົບເວົ້າໝາຍ ຫຼືວ່າບໍ່ມີໄນດ້ເຫຼືອຢູ່ໃນ OPEN
 - ເລືອກໄນດ້ທີ່ດີທີ່ສຸດໃນ OPEN
 - ສ້າງໄນດ້ລູກໃຫ້ກັບໄນດ້ທີ່ດີທີ່ສຸດນັ້ນ
 - ສໍາລັບໄນດ້ລູກແຕ່ລະຕົວໃຫ້ທຳດັ່ງຕໍ່ໄປນີ້:
 - ຖ້າໄນດ້ນັ້ນຍັງບໍ່ເຄີຍຖືກສ້າງມາກ່ອນໜັນນັ້ນ ໃຫ້ກວດສອບຄ່າຂອງມັນໂດຍໃຊ້ຮືວໃຈ ສະຕິກັງຊັນແລ້ວເພີ່ມເຂົ້າໄປໃນ OPEN ແລ້ວບັນທຶກວ່າເປັນໄນດ້ແມ່
 - ຖ້າໄນດ້ນັ້ນຖືກສ້າງມາກ່ອນໜັນນີ້ແລ້ວ ໃຫ້ປ່ຽນໄນດ້ແມ່ຂອງມັນ ຖ້າເສັ້ນຫາງໃໝ່ທີ່ໄດ້ກວ່າໄນດ້ແມ່ຕົວເດີມ ໃນກໍລະນີນີ້ ໃຫ້ປ່ບປ່ຽນຄ່າຕາມເສັ້ນ ຫາງທີ່ອາດຈະເກີດຂຶ້ນ

2.3.4. ອານກຳລິດທຶນ A*

ການຄົ້ນຫາແບບ A* ເປັນອີກແບບຂອງການຄົ້ນຫາແບບດີທີ່ສຸດກ່ອນ ວິທີການເລືອກໄນດ້ທີ່ຈະໃຊ້ໃນການດຳເນີນການຕໍ່ຈະພິຈາລະນາຈາກໄນດ້ທີ່ດີທີ່ສຸດ ແຕ່ໃນກໍລະນີຂອງ A* ມີຈະມີລັກສະນະ ພິເສດກວ່າຄື ໃນສ່ວນຂອງຮືວໃຈສະຕິກັງຊັນໃນກໍລະນີຂອງການຄົ້ນຫາແບບດີທີ່ສຸດກ່ອນນັ້ນຄ່າທີ່ໄດ້ຈາກຮືວໃຈສະຕິກັງຊັນຈະເປັນຄ່າທີ່ໄດ້ຈາກ ໂນດບັດຈຸບັນ ແຕ່ໃນກໍລະນີຂອງ A* ຄ່າຂອງຮືວໃຈສະຕິກັງຊັນ ຈະວັດຈາກຄ່າ 2 ຄ່າຄື ຄ່າທີ່ແກງຈາກໄນດ້ບັດຈຸບັນໄປຍັງໄນດຮາກ ແລະ ຈາກໄນດ້ບັດຈຸບັນໄປຍັງໄນດເວົ້າໝາຍ ຖ້າເຮົາໃຫ້ຕົວປ່ຽນ f ແກ່ນຄ່າຂອງຮືວໃຈສະຕິກັງຊັນ g ເປັນພັ້ງຊັນທີ່ໃຊ້ວັດຄ່າ Cost ຈາກສະຖານະເລີ່ມຕົ້ນຈົນຮອດສະຖານະບັດຈຸບັນ h' ເປັນພັ້ງຊັນທີ່ໃຊ້ວັດຄ່າ Cost ຈາກສະຖານະບັດຈຸບັນຮອດສະຖານະເວົ້າໝາຍດັ່ງນັ້ນ

$$f = g + h'$$

- ອານກຳລິດທຶນແບບ A*(A* algorithm)

ໃນໄນດ້ໜຶ່ງໆຂອງການຄົ້ນຫາແບບ A* ຈະມີອົງປະກອບດັ່ງຕໍ່ໄປນີ້

- ពិវិសាទិបាយ (Description) ថ្លែសម្រាប់នៅលើក្នុងគិតបាយ
 - ឯកសារតំបន់ (Indicator) ថ្លែសម្រាប់គ្មានភាពរបស់ខ្លួននៅលើក្នុងគិតបាយ
 - ពិវិសាទិ (Pointer) ថ្លែសម្រាប់ពាក្យជាប្រភពនៃក្នុងគិតបាយ
 - លិត (List) នៅលើក្នុងគិតបាយដូចជាអាជីវកម្ម

ຕົວ Parent-Link ຈະເປັນສ່ວນທີ່ເຮັດໃຫ້ເກີດການ Recover ຂອງເສັ້ນທາງໄປຍ້ງເປົ້າ
ໝາຍເມື່ອພິບລິດຂອງໂນດລູກທີ່ສະແດງເສັ້ນທາງທີ່ດີທີ່ສຸດ

ສໍາລັບໄນດລູກໜຶ່ງຈະຖືກຈັດອອກເປັນ 2 ປະເທດຕີ:

- ການເຮັດວຽກຂອງອານົາມກຳນົດທຶນ

- ໃຫ້ຄຳນວນຄ່າ $g(\text{SUCCESSOR}) = g(\text{BESTNODE}) + \text{Cost}$ ຈາກ BESTNODE ຮອດ SUCCESSOR
- ໃຫ້ກວດວ່າຄ່າ Cost ຂອງ SUCCESSOR ນີ້ມີ Cost ເທົ່າກັບ OPEN ໂນດອື່ນ ຫຼື ບໍ່ຖືເກົ່າໃຫ້ເອີ້ນໄນດ້ນີ້ວ່າ OLD ແລະ ໃຫ້ຕັດ SUCCESSOR ນີ້ອອກ ແລະ ໃຫ້ OLD ນີ້ຢູ່ໃນລິດ ຂອງ BESTNODE ຂອງໄນດລູກແລ້ວຕັດສິນວ່າ Parent Link ຂອງ OLD ຈະ ຕ້ອງຮີເຊດໃຫ້ຊື່ໄປທີ່ BESTNODE ຫຼືບໍ່ ລົງນີ້ຈະຕ້ອງຮີເຊດຖືເສັ້ນທາງໃໝ່ທີ່ໄດ້ຄົດແລ້ວດີທີ່ສຸດ (ຫຼັງນີ້ເພາະ SUCCESSOR ແລະ OLD ກຳຄົນໄນດ່ງວກັນ) ດັ່ງນັ້ນເຮົາຈຶ່ງຕ້ອງຄົດວ່າເສັ້ນທາງຈາກ Parent ປັດຈຸບັນຮອດ SUCCESSOR ກັບເສັ້ນທາງຈາກ BESTNODE ຮອດ SUCCESSOR ອັນໄດ້ກວ່າກັນ ໂດຍການປົງບໜູບຄ່າ g ຖ້າ OLD ດີກວ່າຫຼືດີເກົ່າກັນ ກຳບໍ່ຕ້ອງຮັດຫຍັງ ຖ້າ SUCCESSOR ດີກວ່າໃຫ້ຮີເຊດ Parent ຂອງ OLD ໃຫ້ລົງຊື່ໄປທີ່ BESTNODE ແລະ ບັນທຶກເສັ້ນທາງທີ່ດີທີ່ສຸດໃນ $g(\text{OLD})$ ແລະ Update $f(\text{OLD})$
- ຖ້າ SUCCESSOR ບໍ່ໄດ້ເປັນ OPEN ໃຫ້ເປົ່ງວ່າເປັນ CLOSE ຫຼືບໍ່ ຖ້າເປັນໃຫ້ເອີ້ນວ່າ OLD ແລ້ວລວມ OLD ນີ້ໃຫ້ຢູ່ໃນລາຍການຂອງ BESTNODE's Successor ກວດສອບວ່າເສັ້ນທາງໄດ້ທີ່ສຸດ (ລະຫວ່າງ ໃໝ່ ແລະ ເກົ່າ) ດ້ວຍວິທີຄູງວກັບຫົວຂໍຂ້າງເທິງ ແລະ ເຊດ Parent ລົງຄ່າ g ແລະຄ່າ f ທີ່ເໝາະສົມ, ໃນກໍລະນີທີ່ OLD ເປັນເສັ້ນທາງທີ່ດີກວ່າເຮົາຕ້ອງເຄືອນ (Propagate) ບຸກຢ່າງໄປທີ່ໄນດລູກຂອງ OLD ໃຫ້ OLD ຊື້ໄປທີ່ SUCCESSOR ຂອງມັນ ແລະ ໂນດລູກແຕ່ລະຕົວຈະຕ້ອງຊື່ໄປຍັງໄນດລູກຂອງມັນເອງ ອີກ ຈົນກວ່າຮອດໄນດສຸດທ້າຍ (Terminal Node) ຂອງແຕ່ລະກົງບໍ່ວ່າໄນດນັ້ນຈະ OPEN ຫຼື ວ່າບໍ່ມີໄນດລູກໃນ ການໄສ່ (Propagate Cost) ໃໝ່ໃຫ້ກັບໄນດຕ່າງໆຂອງໄນດລູກນີ້ຈະຮັດແບບ Depth-First Traverse ເທິງ Tree ເລີ່ມຈາກ OLD ບຸ່ນຄ່າ g ຂອງແຕ່ລະໄນດ (ແລະ ຄ່າ f) ໃຫ້ຢຸດການ Traverse ໃນທຸກກິ່ງທີ່ບໍ່ມີໄນດ ຫຼື ໄນດທີ່ມີຄ່າເສັ້ນທາງທີ່ເກົ່າກັບ ຫຼື ດີກວ່າຫັງໝົດທີ່ໄດ້ຫາມາໃນ Parent Link ຂອງແຕ່ລະໄນດທີ່ຊີ້ກລັບໄປຍັງ Parent ທີ່ດີທີ່ສຸດເກົ່າທີ່ຮູ້ ໃນຂະນະທີ່ເຄື່ອນລົງໄປທີ່ໄນດໃຫ້ກວດວ່າ Parent ຊື້ໄປທີ່ໄນດທີ່ຫາກໍຜ່ານມາ ຫຼືບໍ່ ຖ້າແມ່ນໃຫ້ເຄື່ອນຕໍ່ ຖ້າບໍ່ແມ່ນສະແດງວ່າຄ່າ g ຂອງມັນມີຜົນຕໍ່ເສັ້ນທາງທີ່ດີທີ່ສຸດທີ່ຫາກໍຜ່ານມາ ດັ່ງນັ້ນການເຄື່ອນຈະຢຸດບ່ອນນີ້ ແຕ່ວ່າເປັນໄປໄດ້ທີ່ຄ່າໃໝ່ຂອງ g ຈະຖືກເຄື່ອນລົງເສັ້ນທາງ ທີ່ເຮົາກໍລັງທຳຍູ່ນີ້ ອາດຈະດີກວ່າເສັ້ນທາງທີ່ຜ່ານ Current Parent ໃຫ້ ປົງບໜູບທັງສອງອັນ ຖ້າຫາກວ່າທີ່ Current Parent ດີກວ່າໃຫ້ຢຸດເຄື່ອນ ຖ້າເສັ້ນທາງທີ່ເຮົາກໍລັງທຳຍູ່ດີກວ່າໃຫ້ຮີເຊດ Parent ແລະ ເຄື່ອນທີ່ຕໍ່ໄປ.
- ຖ້າ SUCCESSOR ບໍ່ໄດ້ເປັນຫັງ OPEN ແລະ CLOSE ໃຫ້ OPEN ແລະລວມມັນເຂົ້າໃນລິດຂອງໄນດລູກຂອງ BESTNODE ແລະຄຳນວນ

$$f(\text{SUCCESSOR}) = g(\text{SUCCESSOR}) + h'(\text{SUCCESSOR})$$

ຈາກຕົວຢ່າງການຫາຄຳຕອບດ້ວຍການຄົ້ນຫາຂໍ້ມູນແບບ A*ຂອງປັນຫາ 8-Puzzle ໃນການຄຳນວນຄ່າຂອງ g ຄືການປູ່ບໍ່ທຸກສະຖານະບັດຈຸບັນກັບສະຖານະເລີ່ມຕົ້ນແລະຄ່າ h' ຄືການປູ່ບໍ່ທຸກສະຖານະບັດຈຸບັນກັບສະຖານະເປົ້າໝາຍເຊິ່ງໃນການຄຳນວນຄ່າ g ແລະ h' ຈະມີວິທີການດັ່ງນີ້

ສະຖານະເລີ່ມຕົ້ນ

2	8	3
1	6	4
7		5

ສະຖານະເປົ້າໝາຍ

1	2	3
4	5	6
7	8	9

ສະຖານະບັດຈຸບັນ

2		3
1	8	4
7	6	5

ສະຖານະເປົ້າໝາຍ

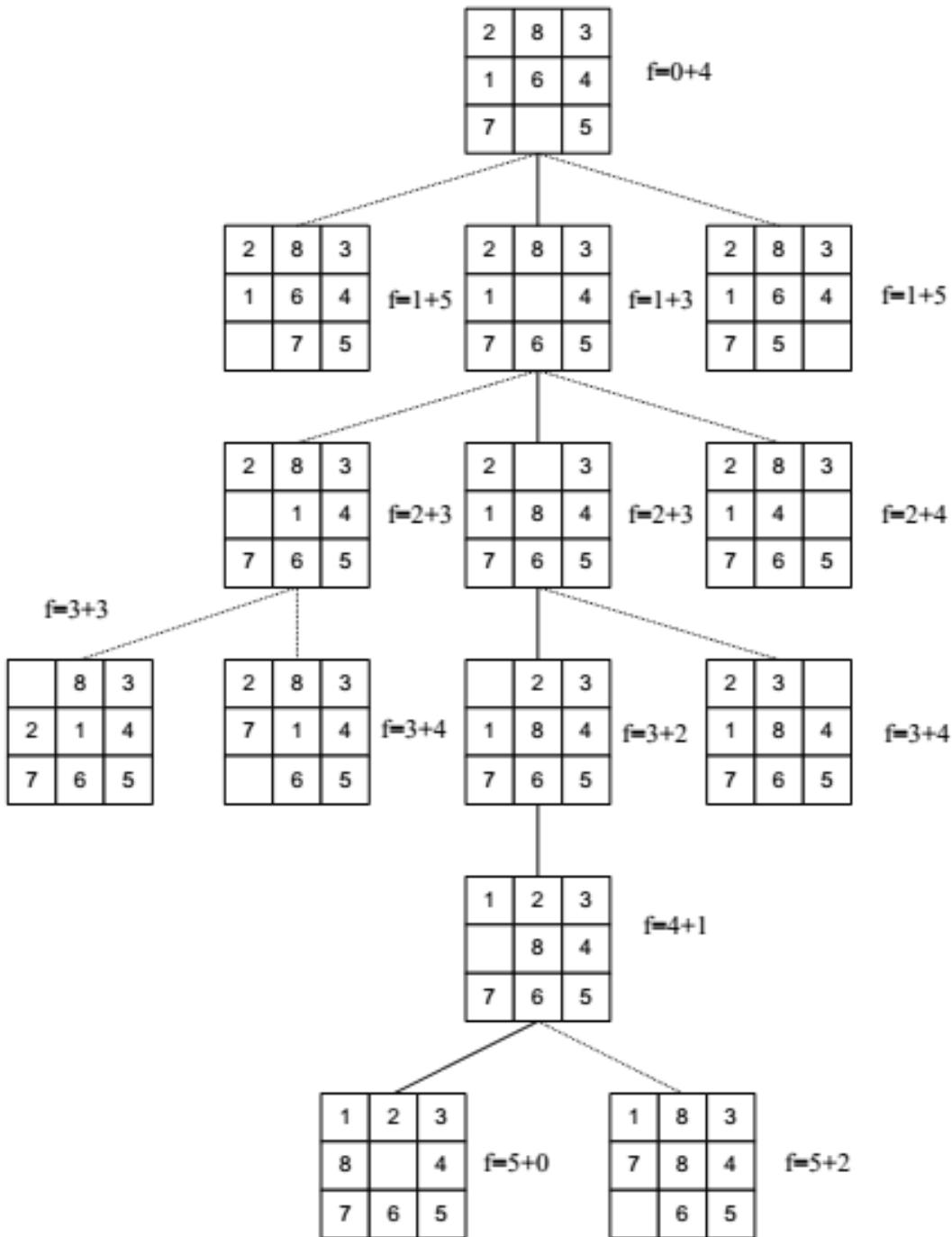
ການຫາຄ່າຂອງ g ເຮັດໄດ້ໂດຍການປູ່ບໍ່ທຸກສະຖານະບັດຈຸບັນກັບສະຖານະເລີ່ມຕົ້ນ ຄື ການນັບວ່າແຕ່ລະບ່ອງຂອງຕາຕະລາງບັດຈຸບັນຫ່າງຈາກຈາກຊຸດເລີ່ມຕົ້ນໄປຈັກປ່ອງ ຄືບ່ອງທີ 1 ຕາຕະລາງທັງສອງບໍ່ຫ່າງຈາກກັນ ດັ່ງນັ້ນໄດ້ຄ່າເປັນ 0 ບ່ອງທີ 1 ບ່ອງທີ 2 ສະຖານະບັດຈຸບັນບໍ່ມີຄ່າບໍ່ຄົດ ແລະຄືກັນທີບ່ອງທີ 3 ແລະ 4 ກຳມີຄ່າເປັນ 0 ເພາະ ທັງສອງຕາຕະລາງບໍ່ຕ່າງກັນ ສຳລັບຕາຕະລາງທີ 5 ຈະມີຄ່າເປັນ 1 ເພາະ 8 ຫ່າງຈາກທີ ທີ່ເຕີຍຢູ່ 1 ບ່ອງ ເຮັດແບບນີ້ເລື້ອຍໆ ເຮົາຈະໄດ້ຄ່າຕາມປ່ອງຕາຕະລາງຕ່າງໆ ດັ່ງນີ້

$$1=0, 2=0, 3=0, 4=0, 5=1, 6=0, 7=0, 8=1, 9=0 \text{ ລວມ } g=2$$

ສຳລັບຄ່າຂອງ h' ຈະໄດ້ດັ່ງນີ້ 1

$$=1, 2=0, 3=0, 4=1, 5=1, 6=0, 7=0, 8=0, 9=0 \text{ ລວມ } h'=3$$

$$f = 2 + 3 = 5$$



ຮູບທີ 2.17 ແກ້ປັນຫາ 8 ປິດສະໜາ ດ້ວຍ A*

○ ຂຶ້ສັງເກດ

1) ກ່ຽວກັບຕົວອານກຳລິດທີມ

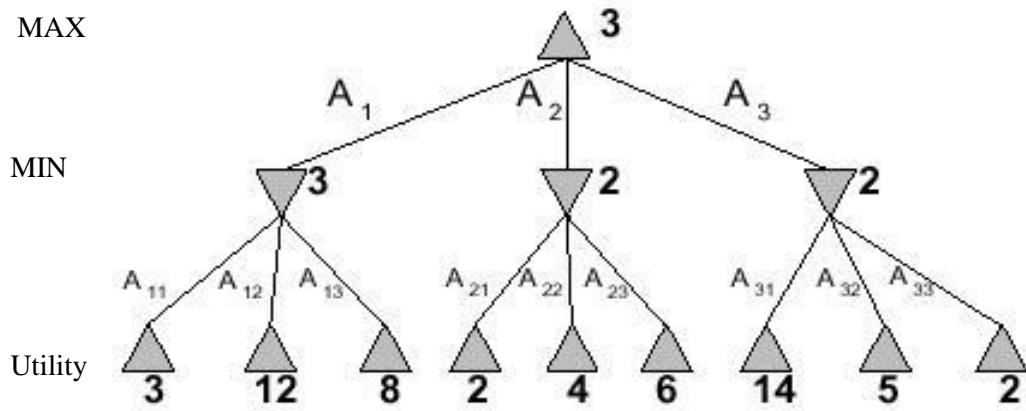
- ໝາຍຄວາມວ່າເຮົາກຳລັງຄຳນິ້ງຫາເສັ້ນທາງທີ່ດີທີ່ສຸດ
- ໃນກຳລະນີທີ່ເຮົາຄຳນິ້ງສະເພາະວ່າຈະເຮັດແນວໄດ້ທີ່ຈະໄດ້ຄໍາຕອບເຖິງນັ້ນເຮົາສາມາດກຳນົດໃຫ້ $g=0$ ໄດ້

- ຖົາຕ້ອງການໃຫ້ໄດ້ເສັ້ນທາງທີ່ມີຂັ້ນຕອນໜ້ອຍທີ່ສຸດ ເຮົາສາມາດກຳນົດໃຫ້ຄ່າຂອງໂນດໄປໜາ
ໂນດລູກມີຄ່າຄົງທີ່ (ເຫົ່າກັບ1)
- 2) ກ່ຽວກັບຄ່າຂອງ h' ແລະ h (ຄ່າຈາກໂນດອດໂນດລູກ)
- ຖົາ h' ເປັນ perfect estimator ຂອງ h , A^* algorithm ກໍຈະກາຍເປັນການເຂົ້າຫາເວົ້າ
ໝາຍໄດ້ຍິ່ນມີການຄື້ນຫາ
 - ເມື່ອ $h' = \text{ຄ່າທີ່ດີທີ່ສຸດ}$ ກຳຄົນການຫາກຳນົດໄລຍະທາງທີ່ໄກ້ທີ່ສຸດ
 - ເມື່ອ $h' = 0$ ການຄື້ນຫາຈະເຖິງຄວບຄຸມໄດ້ g
 - ຖົາ $g=0$ ອີກການຄື້ນຫາຈະເປັນແບບ Random
 - ຖົາ g ເປັນ 1 ການຄື້ນຫາຈະເປັນ Breadth-First

2.4. MINIMAX ALGORITHM

ມີການເຮັດວຽກແບບ Depth-first search ຕີ່ ການສ້າງໂນດລົງໄປທາງເລີກກ່ອນ ຈິນຮອດລະດັບ
ທີ່ຕ້ອງການ (ອາດຈະເປັນຈຸດສິນສຸດຂອງແນວ ຫຼື ຈຸດລະດັບ N ໃດໜຶ່ງ) ໃຊ້ Utility Function ຫຼື
Evaluation function ເພື່ອຫາຄ່າຄວາມດີຂອງໂນດນັ້ນ ສົ່ງຄ່າທີ່ໄດ້ກັບຂຶ້ນໄປໃນລະດັບທີ່ຢູ່ທາງເທິງເພື່ອ
ຕັດສິນໃຈຕໍ່ໄປ ໃຊ້ເຕັກນິກການເຂັ້ນໃຊ້ໂຕເອງ (Recursive) ໄດ້ ຕີ່ Algorithm ຕໍ່ໄປນີ້

- | |
|--|
| <ol style="list-style-type: none"> 1. ສ້າງໂຄງຮ່າງຄື້ນໄມ້ ທັງໝົດລົງໄປໜາ terminal node 2. ໃຊ້ utility function ເພື່ອຫາຄ່າຂອງໂນດນັ້ນ 3. ສົ່ງຄ່າຜົນຮັບກັບໄປໜາໂນດພໍ່ ຕາມກົດດັ່ງນີ້ <ul style="list-style-type: none"> ຖົາໂນດພໍ່ເປັນຜູ້ຫຼືນ (MAX) ຈະຄື້ນຄ່າ Maximum ຂອງ ໂນດລູກ ຖົາໂນດພໍ່ເປັນຜູ້ຫຼືນຜ່າຍກົງຂ້າມ (MIN) ຈະຄື້ນຄ່າ Minimum ຂອງໂນດລູກ 4. MAX ຈະຖືກເລືອກ ໂນດທີ່ໃຫ້ຄ່າທາງໄປທີ່ສູງ (Maximum) ທີ່ສຸດ |
|--|

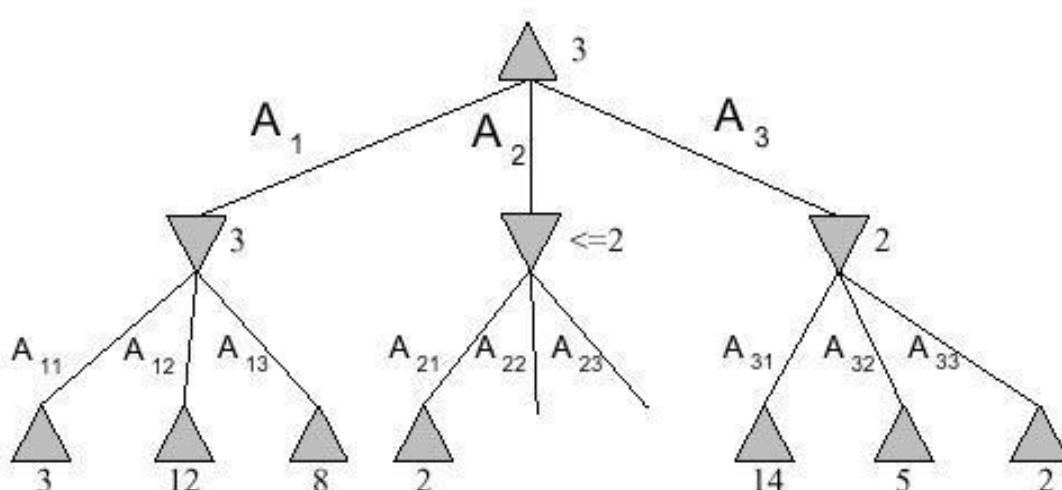


ຮູບທີ 2.18 ສະແດງເສັ້ນທາງ Minimax

Utility function ໃຫ້ຄ່າຄວາມດີທີ່ terminal node ຈາກນັ້ນ ຫຼືລະດັບ A_1 ເຊິ່ງເປັນ MIN ຈະເລືອກຄ່າທີ່ນອຍທີ່ສຸດ ສ່ວນໂນດ MAX ເທິງສຸດຈະສົງຄ່າໄນດລູກໃຫ້ຄ່າໜ້າຍທີ່ສຸດ

2.5. Alpha-Beta Cutoffs

ບັນລຸວ່າ Alpha-Beta Pruning ຕີການຕັດສາຂາທີ່ບໍ່ມີຜົນຕໍ່ການຕັດສິນໃຈອອກໄປ ເຮັດໃຫ້ຄໍານວນໄວຂຶ້ນ Algorithm ນີ້ປັບປຸງຈາກ Minimax Algorithm ໂດຍເພີ່ມການເຮັດວຽກສໍາລັບເລືອກບໍ່ຄໍານວນບາງກິ່ງ ຄ່າ a ແລະ b ທີ່ກາເພີ່ມລົງໄປແຕ່ລະໂນດທີ່ກຳລັງຄົ້ນຫາ ຄ່າ a ເວັບຄ່າທີ່ດີທີ່ສຸດຂອງ MAX ໂນດທີ່ເຄີຍເຫັນມາ ໂດຍທີ່ຄ່ານີ້ຈະບໍ່ມີຫາງໝູດລົງ (MAX ຕ້ອງການເລືອກຄ່າທີ່ໜ້າຍທີ່ສຸດ) ຄ່າ b ເວັບຄ່າທີ່ດີທີ່ສຸດຂອງ MIN ໂນດທີ່ເຄີຍເຫັນມາ ໂດຍທີ່ຄ່ານີ້ຈະບໍ່ມີຫາງເພີ່ມຂຶ້ນ (MIN ຕ້ອງການເລືອກຄ່າທີ່ນອຍທີ່ສຸດ)



ຮູບທີ 2.19 ສະແດງເສັ້ນທາງ Alpha-Beta Cutoff

- វិធានខែង Alpha-Beta Cutoffs

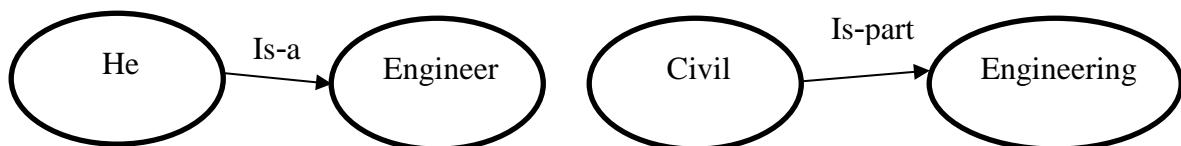
1. ការតើន្យាបានរួចរាល់ពីលទ្ធផលដែលមិនអាចបង្កើតឡើង ខាង MIN មើលតា $\beta \leq A$ ខាងលើលទ្ធផល MAX នឹងបង្កើតឡើង

2. ការតើន្យាបានរួចរាល់ពីលទ្ធផលដែលមិនអាចបង្កើតឡើង ខាង max មើលតា $\alpha \geq B$ ខាងលើលទ្ធផល MIN នឹងបង្កើតឡើង

ບົດທີ 3 ການສະແດງຄວາມຮູ້ (knowledge Representation)

3.1 Semantic Network

Semantic Network เป็นรูปแบบภาษาศาสตร์ความรู้โดยใช้กราฟ มาช่วยในการสืบความหมาย, Semantic network ที่กันเดียวกันโดยตัวล้วน (Quillian, 1968) เพื่อใช้แทนความหมายของคำในภาษาอังกฤษ, โดยการแทนข้อความ และ ข้อมูลด้วยโนดที่เชื่อมต่อกันโดยອາກ (ลิง) เช่น อາກแต่ละอันจะลงทะเบียนให้เป็นกราฟเพิ่มความสำนึกร่องโนดแต่ละโนดที่เชื่อมต่อกัน.

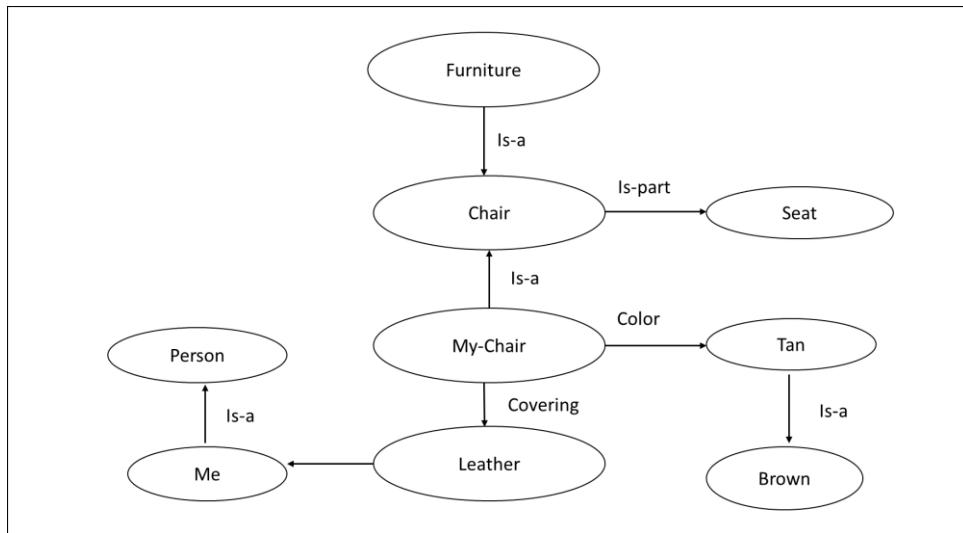


ຮັບທີ 3.1 ຄວາມຮັ້ງຂອງປະໂຫຍກ He is an engineer ແລະ Civil is part of engineering

ຈາກຕົວຢ່າງໃນຮູບທີ 3.1 ສະແດງໃຫ້ເຫັນວິທີການສະແດງຄວາມຮູ້ແບບ Sematic Net ຂອງປະໂຫຍກ ‘ He is an engineer’ ແລະ “Civil is part of engineering” ເຊິ່ງປະກອບດ້ວຍໄນດຂອງ He, Engineer, Civil Engineering ທີ່ຖືກເຊື້ອມຕໍ່ໂດຍ ອາກຂອງ is-a ແລະ is part ສະແດງຄວາມສຳພັນຂອງ He ແລະ engineer ກັບ Civil ແລະ engineering ຕາມລຳດັບ.

ສຳລັບນິຍາມຂອງໂນດ ແລະ ອາກ ຍັງບໍ່ມີການກຳນົດທີ່ແນ່ນອນ, ເຖິງຢ່າງໄດ້ກຳຕາມ
ການສະແດງຄວາມຮັບແບບນີ້ ຈະມີຫຼັກການບາງຢ່າງທີ່ຮັກຈຳກຳໄດ້ຍົກເວລີໄປດ້ານນີ້

- ໂນດ(Node) ໃຊ້ສະແດງເຖິງ ຂື່ຂອງວັດຖຸ (Object), ຄຸນສົມບັດ, ເຫດການ, ການເຮັດວຽກ
 - Object (ວັດຖຸ) ເຊັ່ນ Bird, Car, Basket-ball, Water, Glass, Cup, Table
 - Attribute ໃນນີ້ໝາຍເຖິງຄຸນສົມບັດຂອງວັດຖຸ ເຊັ່ນ Red, 20, good
 - Situation (ເຫດການ) ເປັນຄໍານາມທີ່ອະທິບາຍເຫດການ ເຊັ່ນ: Business-Crisis, Dead
 - Action (ການເຮັດວຽກ) ເຊັ່ນ Give, Go, Fly
 - ອາກ (Arc) ສະແດງຄວາມໝາຍຂອງຄວາມສຳພັນລະຫວ່າງໂນດ. ບໍ່ມີກົດການຂຽນ Arc ຕີ່ຈະ
ຂຽນຄໍາຫຍັງກໍໄດ້ທີ່ສີເຕິງ Node ທີ່ສຳພັນກັນແຕ່ທີ່ນີ້ຢືນໃຊ້ໄດ້ແກ່ Is-a, Has-part, Instance-of
ທີ່ ສືບ່ານ



ຮູບທີ 3.2 ການສະແດງຄວາມຮູ້ໃນຮູບທີ 6.2 ເປັນການສະແດງຄວາມຮູ້ໃນຮູບແບບຂອງການປັບປຸງ
ການສະແດງຄວາມຮູ້ຈຶ່ງໃນຄອມພິວເຕີນັ້ນ ເຮົາຈະໃຊ້ຮູບແບບຂອງຕັກກະສາດແບບໂປລັອກ
(PROLOG) ແລະ ລືບ (LISP) ດັ່ງຕົວຢ່າງ ຕາຕະລາງ 3.1 ແລະ ຮູບທີ 3.3

Atom	Property List
Chair	((is-a Furniture))
My-Chair	((is-a Chair) (color Tan) (covering Leather) (owner Me))
Me	((is-a Person))
Tan	((is-a Brown))
Seat	((is-part Chair))

ຕາຕະລາງ 3.1 ການສະແດງຄວາມຮູ້ໂດຍໃຊ້ ລືບ (LISP)

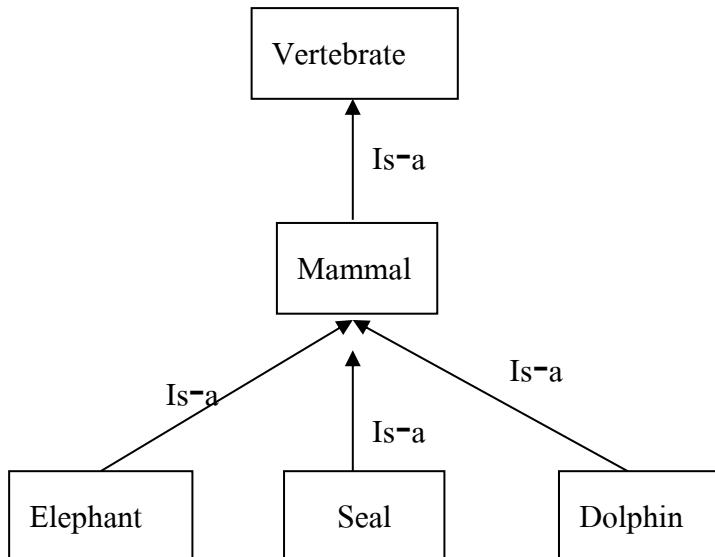
Is-a (Chair, Furniture)
Is-a (Me, Person)
Covering (My-Chair, Leather)
Color (My-Chair, Tan)

ຮູບທີ 3.3 ການສະແດງຄວາມຮູ້ໄດຍໃຊ້ເພດີເຄີດໄລຈິກ

3.1.1. Taxonomical Hierarchy

ເປົ້າຂະບວນການໃນການຈັດລຳດັບຂັ້ນ(Hierarchy) ດັ່ງທີ່ສະແດງໃນຮູບ 3.4 ເຊິ່ງສະແດງການຈັດ class ຕ່າງຂອງ mammal ສ່ວນທີ່ມີຂັ້ນທີ່ສູງກວ່າເئັນໄດ້ວ່າ Superclass ຕີ່ Vertebrate ແລະ ຂັ້ນຕໍ່ກວ່າ ຫຼື Class ຍ່ອຍຕີ່ Elephant, Seal ແລະ Dolphin ການເຊື້ອມຂອງໂນດຫັງໝົດຈະເຫັນວ່າອາກມີຄວາມສໍາພັນເປັນ is-a ເນື້ອເຮົາຊຽນໃນຮູບແບບຂອງເພດີເຄີດໄລຈິກ ຈະໄດ້ດັ່ງລຸ່ມນີ້:

Is-a (Mammal, Vertebrate)
is-a (Elephant, Mammal)
Is-a (Seal, Mammal)
Is-a (Dolphin, Mammal)



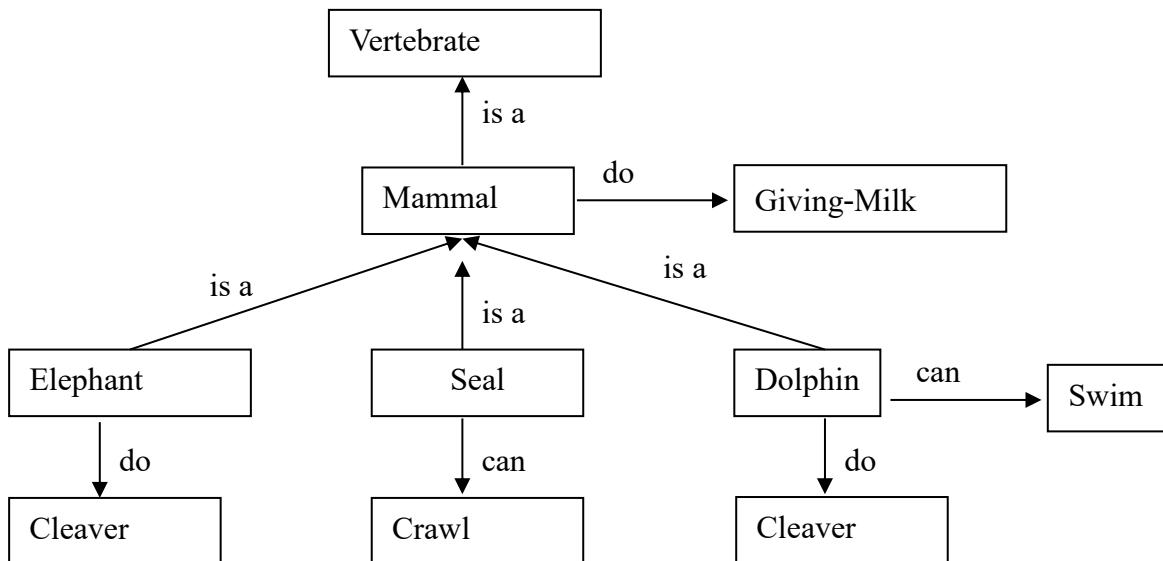
ຮູບທີ 3.4 Sematic network

ສໍາລັບ Taxonomical Hierarchy ສາມາດໃສ່ຄຸນສົມບັດຂອງໂນດເຂົ້າໄປນໍາ ເຊັ່ນ:

do (Mammal, Giving-Milk)
do (Elephant, Clever)
do (Dolphin, Clever)

- can (Elephant, Walk)
- can (Seal, Crawl)
- can (Seal, Swim)
- has (Mammal, Warm-blood)

ດັ່ງນັ້ນການສະແດງຄວາມຮູ້ດ້ວຍ Semantic Network ສາມາດຂະຫຍາຍໄດ້ດັ່ງຮູບທີ 3.5



Somatic Networks മേഖലയുമ്പിട്ടോ, 2, എല്ലാം

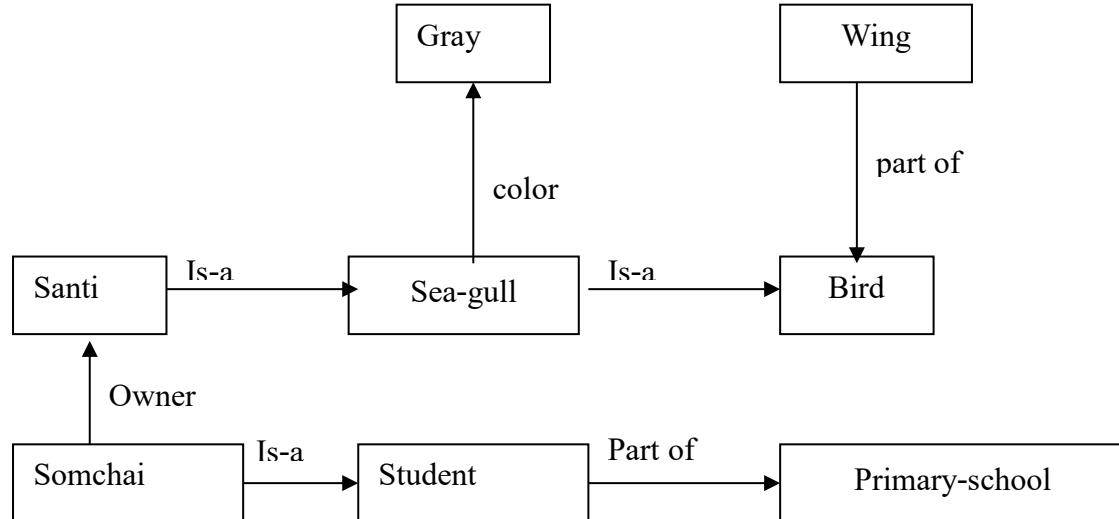
- Transitivity หมายความว่า ถ้า Y อยู่ใน X และ Z อยู่ใน Y แล้ว Z อยู่ใน X
 - is-a (X,Y) ^ is-a (Y,Z) \rightarrow is-a (X,Z)
is-a (Elephant, Mammal) ^ is-a (Mammal, Vertebrate) \rightarrow is-a (Elephant, Vertebrate)
 - has-part (X, Y) ^ has-part (Y,Z) \rightarrow has-part (X,Z)
has-part (Body, Hand) ^ has-part (Hand, Finger) \rightarrow has-part (Body, Finger)
 - การสืบทอดคุณสมบัติ (Property inheritance) คือ การที่ Object ที่อยู่ในคลาสพ่อ ได้รับคุณสมบัติมาโดยอัตโนมัติ
Is-a (Elephant, Mammal)
Do(Mammal, Giving-Milk)
จะได้ Do(Elephant, Giving-Milk)

3.1.2. ភាពមាត្រាបេណ្ឌជិំតុលិនិខទាំងមួយ Semantic Network

ການຫາເຫດຜົນຂອງຄວາມຮູ້ແບບນີ້ສາມາດເຮັດໄດ້ຈ່າຍໆ ໂດຍການຄົ້ນຫາຈາກເສັ້ນເຊື່ອມ ຫຼື ອາກຂອງຖານຄວາມຮູ້ນັ້ນ ແລ້ວສ້າງການປ່ຽນທຸກຈາກໂນດທີ່ຕີກັນເພື່ອປ່ຽນທຸກຫາຄໍາຕອບ. ຫຼື ເວົ້າ

ໄດ້ອີກແບບໜຶ່ງວ່າ ການຫາເຫດຜົນຂອງ Semantic Network ອາໄສຫຼັກການພື້ນຖານຂອງການປັບປຸງບຮູບແບບ (Pattern Matching) ແລະ ການຖໍາຍທອດຄຸນສົມບັດ (Property Inheritance)

ຕົວຢ່າງ ການບະສະດູຄວາມຮູ້ແບບ sematic network ຮູບທີ 3.6 ຖ້າເຮົາຄ້ອງການຮູ້ວ່າ Santi ເປັນສີຫຍັງ, ການຫາເຫດຜົນຈະເລີ່ມຈາກໂນດທີ່ເປັນ Santi ກ່ອນ ເຊິ່ງເຫັນວ່າ Santi ມີເສັ້ນເຊື່ອມຕໍ່ is-a ທີ່ເຊື່ອມເຂົ້າກັບໂນດ Seagull ທີ່ມີ color ເປັນເສັ້ນເຊື່ອມຕໍ່ ແລະ ເຮັດໃຫ້ຮູ້ວ່າ Santi ມີສີ gray. ເນື່ອງຈາກວ່າ Santi ມີ ອາກທີ່ເປັນ is-a ຂອງ seagull ດັ່ງນັ້ນໂນດ Santi ຈຶ່ງໃຊ້ອາກຂອງ color ເພື່ອບອກສືບຂອງ Santi ໄດ້ຕາມກົດການສືບທອດຄຸນສົມບັດ



ຮູບທີ 3.6 ການບະສະແດງຄວາມຮູ້ແບບ Sematic Network

ເພື່ອໃຫ້ເຂົ້າໃຈໝາຍຂຶ້ນເນື້າຈະໃຊ້ ເທື່ໂປໂປລອກ(Turbo Prolog) ຂຸງເພື່ອສະແດງວິທີຫາເຫດຜົນ

/* Program Find what is the color of Santi*/

Predicates

Is-a(string,string)

Owner(string,string)

Partof(string,string)

Color(string,string)

Goal

Write("Question: What is a colorof SANTI", nl.

Color(santi,x)

Write("Anwer: The color of SANTI is",x), nl.

Clauses

Isa(santi,sea_gull).

Isa(sea_gull,bird).
Isa(Marcus,sea_gull).
Isa(somchai,student).
Isa(x,y) if isa(x,z) and isa(z,y)..
Color(seal_gull,gray).
Color(x,y) if isa(x,z) and isa(z,y).
Part of(wing,bird).
Part of(student,primary_school).
Part of(x,y) if part of(x,z) and isa(y,z).
Owner(somchai,santi).

ເມື່ອ compile ໂປຣແກຣມແລ້ວ Run ຈະໄດ້ຜົນຮັບດັ່ງນີ້

A:>Santi

Question: What is a color of Santi

Answer: The color of Santi is gray

3.2. Frame

ວິທີການສະແດງຄວາມຮູບແບບ ເຟຣມ (Frame) ຕີກຝັດທະນາໂດຍ ມິ້ນສະກິ (Minsky, 1975) ເປັນການສະແດງຄວາມຮູບແບບໜຶ່ງທີ່ຂະໜາຍາຍມາຈາກ semantic network ໂດຍການເພີ່ມຂ່າວສານ (information) ຂອງໂນດໃຫ້ສາມາດບັນຈຸຂໍ້ຄວາມໄດ້ຫຼາຍຂຶ້ນແທນທີ່ຈະມີຢູ່ ອອບເຈັກເຖິ່ງນັ້ນ, ໂດຍ ການສະແດງຄວາມຮູບແບບນີ້ໃຊ້ເຟຣມໃນການອະທິບາຍເຫດການ (Event), ການເຮັດວຽກ (Action) ແລະ ສະຖານະ(Status) ຂອງວັດຖຸ (Object). ເຊິ່ງມີໂຄງສ້າງຂຶ້ນນັ້ນນີ້:

```
<Frame> ::=           <Frame Name>
                  <Slot Name> <Slot Value>
                  ...
                  <Slot Name> <Slot Value>
```

ຕົວຢ່າງການສະແດງຄວາມຮູບແບບເຟຣມ

Mammal

 subclass: Animal

 warm-blooded: yes

Elephant

subclass: Mammal

* color: gray

* size: large

Pi-Yai

instance: Elephant

color: white

owner: Prannoi

Pi-Noi

instance: Elephant

size: small

ຈາກຕົວຢ່າງ <Slot Name> ຫີ້ຂີ້ subclass ແລະ instance ເປັນໂຕເຊື້ອມເພຣມທັງໝົດເຂົ້າຫາກັນ, ເພຣມ Mammal ຈະເປັນເພຣມທີ່ຢູ່ເທິງສຸດ, ມີເພຣມ Elephant ເປັນ class ຍ່ອຍ ຫີ້ class ຍ່ອຍຂອງ Mammal ແລະ ເພຣມ Elephant ມີເພຣມທີ່ມີຂີ້ Pi-Yai ແລະ ເພຣມ Pi-Noi ເປັນເພຣມລູກ.

3.2.1. ການຖ່າຍທອດຄຸນສົມບັດ

ການຖ່າຍທອດຄຸນສົມບັດ (Property Inheritance) ເປັນວິທີການທີ່ເພຣມທີ່ຢູ່ໃນລະດັບຕໍ່ກວ່າ (Subclass) ສາມາດໃຊ້ຄຸນສົມບັດທີ່ຢູ່ໃນເພຣມສູງກວ່າໄດ້. ເປັນຫຼັກການດູວກັບການຖ່າຍທອດຄຸນສົມບັດຂອງ Sematic network, ຈາກຕົວຢ່າງ ເພຣມ Mammal, Elephant, Pi-Yai ແລະ Pi-Noi ຂ້າງເທິງຈະຮູ້ໄດ້ວ່າ Pi-Yai

ເປັນຊ້າງສີຂາວ (ບໍ່ແມ່ນສີເທິງ), ມີຂະໜາດໃຫຍ່ ແລະ ມີເຈົ້າຂອງແມ່ນ Prannoi. ແລະເຫັນໄດ້ວ່າເພຣມ Pi-Yai ຮັບຄຸນສົມບັດເລື່ອງຂອງຂະໜາດມາຈາກເພຣມແມ່ນ (Elephant) ແຕ່ເລື່ອງສີຂາວແມ່ນເປັນຂອງເພຣມ Pi-Yai ເອງ. ໃນທາງດູວກັນ ເພຣມ Pi-Noi ມີສີເທິງ ແລະ ມີຂະໜາດນ້ອຍ; ຍັນເຫດຜົນດູວກັນກັບເພຣມ Pi-Yai ການຖ່າຍທອດຄຸນສົມບັດທີ່ຊັບຊອນຂຶ້ນຄື ສິ່ງທີ່ອອບເຈັກໜຶ່ງມີ Superclass ຫຼາຍກວ່າໜຶ່ງ ແລະ ການຖ່າຍທອດຄຸນສົມບັດເປັນເລື່ອງຂອງການພິຈາລະນາວ່າ ຄຸນສົມບັດຂອງເພຣມໄດ້ທີ່ເໝາະສົມກັບ class ທີ່ກຳລັງພິຈາລະນາຢູ່ດັ່ງນີ້:

Animal

has-part: head

Mammal

subclass: Animal

has-part: nose

warm-blooded: yes

Elephant

```

Subclass: Mammal
has-part: trunk
* color : gray
* size: large

```

Pi-Yai

```

instance: Elephant
color: white
owner: Prannoi

```

ຈາກຕົວຢ່າງຂ້າງເທິງ ເພີ່ມ Pi-Yai ເປັນ class ຍ່ອຍ ຂອງ Elephant ແລະ Elephant ເປັນ class ຍ່ອຍຂອງ Mammal ດັ່ງນັ້ນ Pi-Yai ຈຶ່ງຮັບຄຸນສົມບັດຂອງ Elephant ທີ່ ເປັນຂ້າງສີຂາວ, ມີ ຖົງ (Trunk) ແຫນມີຈະຮັບຄຸນສົມບັດຂອງ Mammal ທີ່ມີແຕ່ດັ່ງ ທຳມະດາ, ເຊິ່ງໃນກໍລະນີນີ້ກິນໄກ ການເລືອກຄືການເລືອກ class ທີ່ຢູ່ຕິດກັນ, ແຫນການເລືອກຄຸນສົມບັດຂອງ class ທີ່ຢູ່ທ່າງອອກໄປ

ໃນບາງກໍລະນີ ສະລັອດ (Slot) ສາມາດຮັບການຖ່າຍຫອດຄຸນສົມບັດໄດ້ຫຼາຍກວ່າ 1 ຄ່າ, ເພື່ອ ປະຫຍດການໃຊ້ພື້ນທີ່ເພີ່ມ, ດັ່ງໃນຕົວຢ່າງຂ້າງເທິງ; ສຳລັບສະລັອດ has-part ຂອງ Mammal ທີ່ມີ Nose ຢູ່ແລ້ວ, ແຕ່ສະລັອດ has-part ຂອງ Animal ກໍ່ມີ has-part ທີ່ເປັນ head ຢູ່ຄືກັນ, ດັ່ງນັ້ນ has-part ຂອງ Mammal ຄວນຈະໄດ້ຮັບຄ່າ Headເຂົ້າມານຳ, ເຊິ່ງໃນຕົວຈີງ has-part ຂອງເພີ່ມ Mammal ຈະຕັອງລະບຸວ່າເປັນຫຼາຍຄ່າ (Multi Value). ໃນຫາງດຽວກັນສະລັອດບາງອັນກໍບໍ່ສາມາດຮັບຫຼາຍຄ່າ ເຊັ່ນ: ສະລັອດ color ຂອງ Elephant ແລະ Pi-Yai ຄວນເປັນຄ່າດຽວ (Single Value).

3.2.2. Slot

ການຂະຫຍາຍສະລັອດ (Slot) ແລະການສະແດງຄວາຮູ້ແບບເພີ່ມນີ້, ສາມາດຂະຫຍາຍຄຸນສົມບັດ ຂອງສະລັອດໄດ້ ແລະ ມີໂຄງສ້າງຂໍ້ມູນດັ່ງນີ້:

```

<Slot> ::= <Slot Name>
          <Slot Name> <Slot Value>
<Slot> ::= <Slot name>
<Facet Name> <Facet Value>
<Facet Name> <Facet Value>

```

.....

```

<Slot> ::= <Slot name>
<Facet Name> <Facet Value>
<Facet Name> <Facet Value>

```

ໃນການຂະໜາຍ **<Slot>** ນີ້ຈະເກີດ Attribute ໃໝ່ທີ່ມີຊື່ວ່າ ພາເຊັດ (Facet) ເຊິ່ງສາມາດແກ່ນດ້ວຍຄຸນສົມບັດ 2 ແບບ ຄື:

1. ພາເຊັດ (Facet) ເປັນວິທີການສະແດງຄວາມຮູ້ແບບກຳນົດຄ່າຕ່າງໆໃນເພື່ອມເຊັນ: Body, Window< Range ຫຼື Default ເປັນຕົ້ນ, ດັ່ງຕົວຢ່າງລຸ່ມນີ້

Frame: CAR

Specialization of: LAND VEHICLE

Body: steel

Window: glass

Mobility: self-propelled

Mobility Mechanism: has wheels

Tires: Rubber

Model:

Range : (sedan, convertible, 2-door, station wagon)

Default: sedan

Fuel:

Range: (gasoline, diesel, propane)

Default: gasoline

Number of seats:

Range: (1-9)

Defult: none

2. Attached Procedure ເປັນຂະບວນການສະແດງຄວາມຮູ້ໂດຍ Procedure, ໂດຍທີ່ເຮົາສາມາດເພີ່ມ Procedure ເຂົ້າໄປເພື່ອຊ່ວຍໃນການເຮັດວຽກຂອງເປັນໃນກໍລະນີທີ່ຕ້ອງການຮູ້ຄ່າເພີ່ມເຕີມ, ແຕ່ບໍ່ມີການສະແດງຄວາມຮູ້ຢູ່ໃນເພື່ອມນັ້ນ. ຂະບວນການທີ່ເພີ່ມຂຶ້ນອາໄສຄຳສັ່ງ ເຊັນ: if-needed Facet, if-added Facet ແລະ if-remove Facet

- if-needed ເປັນ Procedure ທີ່ໃຊ້ຕົ້ນຫາຄ່າໃຫ້ກັບສະລັດເມື່ອຕ້ອງການຮູ້ຄ່ານັ້ນ
- if-added ເປັນ Procedure ທີ່ໃຊ້ສຳລັບຄິດໄລ່ຫາຄ່າທີ່ຕ້ອງການເມື່ອມີຄວາມຈຳເປັນ
- if-remove ເຮັດໜ້າທີ່ກົງກັນຂ້າມກັບ if-added ຕີການລືບຂໍ້ມູນອອກເມື່ອຈຳເປັນ

ຕົວຢ່າງຕໍ່ໄປນີ້ເປັນການສະແດງຄວາມຮູ້ດ້ວຍເພື່ອມທີ່ມີການໃຊ້ Attached Procedure ຈະເຫັນໄດ້ວ່າໃນສະລັດຂອງ Fuel Remaining ທີ່ມີພາເຊັດເປັນ Range ເຊິ່ງຈະບອກລະດັບນໍາມັນຂອງລົດທຸກລະດັບທີ່ຖືກຕ້ອງເມື່ອເວລາຜ່ານໄປ, ແຕ່ບໍ່ຮູ້ວ່າປັດຈຸບັນມີນຳມັນຢູ່ທີ່ໄດ. ໃນກໍລະນີທີ່ຕ້ອງການຮູ້ວ່າ

ສຳລັບ If-needed ຈະເຮັດວຽກເນື້ອມີການຖາມຫາວ່າລົດຄົນນີ້ອາຍຸທີ່ໄດ້ ພາເຊັດທີ່ເປັນ year ຈະບອກຕ່າງປີຂອງປັດຈຸບັນ, ຈາກນັ້ນ If-needed: compute years ຈະເປັນໂຕຄິດໄລ່ໂດຍການນຳປີປັດຈຸບັນໄປລົບບີທີ່ຜະລິດລົດ ຄື ພາເຊັດ year of Manufactured ເຊິ່ງມີຢູ່ໃນເພັມອອກມາເປັນອາຍຸຂອງລົດ, ແລ້ວຄ່າທີ່ໄດ້ເຮັບໄວ້ໃນ <value> ບີຢູ່ທ່າງລຸ່ມ

Frame: CAR

Specialization of: LAND VEHICLE

Body: Steel

Window: Glass

Fuel Remaining:

Range : (empty, $\frac{1}{4}$ tank, $\frac{1}{2}$ tank, full)

Default : none

IF-NEEDED: check fuel gauge

Year of Manufacture:

Year: 1987

Default: none

IF-ADDED: compute years

Value:

ຕົວຢ່າງຂອງ Attached Procedure ໃນນີ້ຄື IF-NEEDED IF-ADDED

3.3. ການເພີ້ງພາດ້ານມະໂນພາບ (Conceptual dependency)

ການເພີ່ມາດ້ານມະໂນພາບ (Conceptual dependency) ຫຼື CD ເປັນທິດສະດີຂອງພາສາທຳມະຊາດ (Natural Language) ແລະ ການປະມວນຜົນພາສາທຳມະຊາດ (Natural Language Processing) ເຊິ່ງຖືກພັດທະນາໄດ້ແຊງ (Schank, 1972) ດ້ວຍຈຸດປະສົງທີ່ຕ້ອງການສ້າງໂປຣແກຣມຄອມພິວເຕີໃຫ້ສາມາດເຮັ້າໃຈພາສາມະນຸດໄດ້ດີ ແລະ ສະຫຼຸບເນື້ອໃນອອກມາແປເປັນພາສາອື່ນໄດ້ ແລ້ວຢ່າງສາມາດຕອບຄໍາຖາມກ່ຽວກັບເລື່ອງເງົ້ານັ້ນໄດ້ເຊັ່ນ: ຖ້າເວົ້າວ່າ “Somchai sold his car.” ຈະເຫັນວ່າໃນປະໂຫຍກນີ້ມີສິ່ງທີ່ເຊື່ອງຢູ່ພາຍໃນຄືເງິນ, ເນື້ອຄົນອ່ານແລ້ວຈະຮູ້ວ່າສິ່ງນີ້ຈະເປັນສິ່ງທີ່ກ່ຽວຂ້ອງຢູ່ພາຍໃນ, ເຊິ່ງບໍ່ໄດ້ມີການເວົ້າເຖິງເລີຍ ຫຼື ຖ້າມີໃຜຖາມວ່າ ດຽວນີ້ສົມຊາຍເປັນເຈົ້າຂອງລົດບໍ່? ເລື່ອງຄອມພິວເຕີໄດ້ກໍ່ຕາມທີ່ສາມາດເຮັ້າໃຈພາສາມະນຸດໄດ້ດີຈະຕ້ອງຕອບວ່າ “ບໍ່” ແລະ ຈະຕ້ອງຕອບວ່າ “ສົມຊາຍໄດ້ຮັບເງິນ” ຖ້າມີການຖາມວ່າ “ສົມຊາຍໄດ້ຮັບເງິນຫຼືບໍ່”

3.3.1. ກົດຂອງການເພີ່ມພາດ້ານມະໄນພາບ

ກົດຂອງການເພີ່ມາດ້ານມະໂນພາບເປັນການສະສົມແນວຄວາມຄິດຕ່າງໆ, ທີ່ເປັນເລືອງຂອງປະສົບການດັ່ງໃນຕົວຢ່າງທີ່ກ່າວມາຂ້າງເຖິງ “Somchai sole his car” ຈາກປະສົບການຈະສາມາດເຊົ້າໃຈໄດ້ວ່າເງິນຈະຕ້ອງເຂົ້າມາກ່ຽວຂ້ອງ ແລະ ກົດຂອງ CD ກໍ່ຈະສະສົມແນວຄວາມຄິດກ່ຽວກັບເລືອງເຫຼົານີ້ໄວ້, ໃນການເວົ້າເຖິງເລືອງເຫຼົານີ້ສິ່ງທີ່ຈະຕ້ອງອ້າງເຖິງສະເໝີຄື Sematic, ເຊິ່ງໃນຄວາມໝາຍຂອງ CD ຈະໝາຍເຖິງ ກົດຂອງປະສົບການທີ່ໃຊ້ສໍາລັບການສະສົມແນວຄວາມຄິດ. ເປົ້າໝາຍຂອງການສະແດງຄວາມຮູ້ໃນລັກສະນະນີ້ຄື:

- ການສະແດງຄວາມຮູ້ເລີ່ມຈາກປະໂຫຍກ
 - ເພື່ອຮັດໃຫ້ປະໂຫຍກສາມາດຫາຂໍ້ສະໜູບໄດ້
 - ເພື່ອຮັດໃຫ້ຄໍາທີ່ໃຊ້ໃນປະໂຫຍກນຳເຂົ້າກັບຄໍາທີ່ໃຊ້ສະແດງຄວາມຮູ້ມີຄວາມເຊື່ອມເຂົ້າກັນໄດຍກົງ
 - ເພື່ອກຳນົດປະໂຫຍກທີ່ຂຽນຕ່າງກັນແຕ່ມີຄວາມໝາຍຄືກັນ, ມີວິທີການສະແດງຄວາມຮູ້ເປັນຮູບແບບ

ចុះវក្សា

- ❖ วิธีการสังเคราะห์ความรู้ของ CD มีขั้นตอนดังนี้:

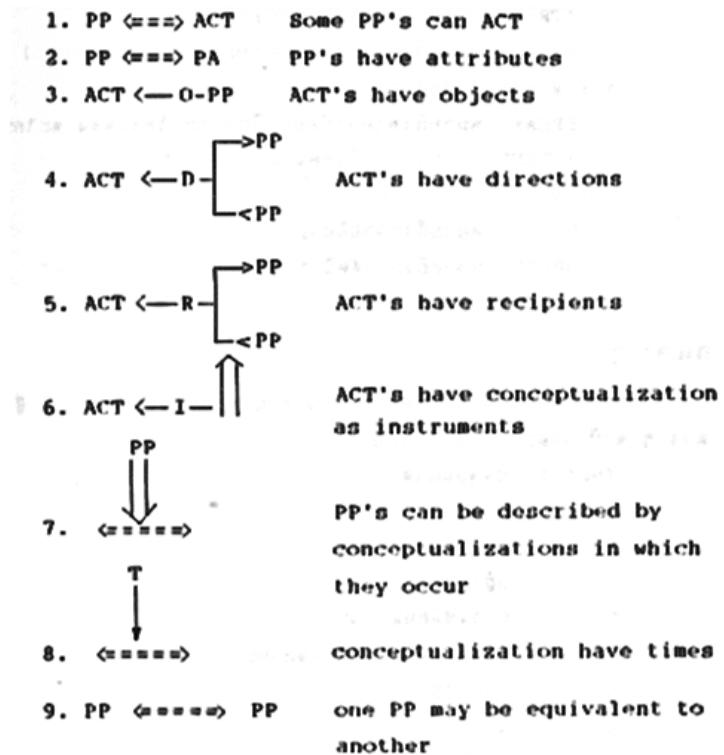
- มีภารกิจที่ต้องดำเนินการในลักษณะขั้นตอนๆ ที่จะดำเนินการตามลำดับ
 - มีขั้นตอนการทำงานที่ซ้ำๆ อยู่บ่อยๆ
 - ไม่ต้องใช้ความคิดสร้างสรรค์มากนัก

- ❖ ឧទតាំស៉ាមាតចាមានីឱ្យក្រុងឯកសារ CD មិត្តភក់នូវប្រព័ន្ធដែលបានរាយការណ៍ដោយខ្លួន

- PPs; Picture Procedures: Object ຕ່າງໆ ວັດຖຸທາງກາຍໃຫຍ່ພາບ: PPs ອາດເປັນຜູ້ເຮັດວຽກ ແລະ ຜູ້ຖືກເຮັດວຽກ (ເຊັ່ນ: ຄົນ, ສິ່ງຂອງ ເປັນຕົ້ນ) ບາງຄັ້ງອາດເປັນຈຸດເລີ່ມຕົ້ນ ຫຼື ຈຸດໝາຍ ປາຍທາງກໍໄດ້. ວັດຖຸທາງມະໂນພາບ: PPs ຈະເປັນຄຸນສົມບັດບາງຢ່າງກໍໄດ້ ເຊັ່ນ ຄວາມຮ້ອນ
 - PAs; attributes of PP's state(Value) ໝາຍເຕີງຄ່າທີ່ບອກລະດັບຂອງ Attribute ເຊັ່ນ ພັ້ນ ເປັນ ລັກສະນະຂອງ Attribute ຂອງສີ ແລະ 5 ແມ່ດ ເປັນລັກສະນະຂອງ Attribute ຂອງຄວາມສູງ
 - ACTs; action ໝາຍເຕີງການເຮັດວຽກຕໍ່ Object ໂດຍຜູ້ເຮັດວຽກ(PP)
 - AAs; Action Aiders ໝາຍເຕີງສິ່ງທີ່ຂະຫຍາຍເພີ່ມເຕີມຊະນິດຂອງ ACT
 - LOCs; Location ໝາຍເຕີງສະຖານທີ່
 - Ts; time ໝາຍເຕີງເວລາ

3.3.2 ກົດຕ່າງໆ ແລະ ໂຄງສ້າງຂອງ CD

ໃນ CD ມີກົດມາດຕະຖານທີ່ຖືກອອກແບບໂດຍແຊງ (Schank, R.C., 1975) ເພື່ອອະທິບາຍແນວຄິດລັດ້ານມະໂນພາບຂອງການເຮັດວຽກໃນຮູບແບບຕ່າງໆໄວ້ ດັ່ງສະແດງໃນຮູບທີ 3.7



ຮັບທີ 3.7 ກົດແລະ ໂຄງສ້າງຂອງ CD

3.3.3 ACT ឧខណិត់ាង្វីន CD

ការងារទី២ ឬ ACT ខាងការនេះបានរាយការណ៍ដែលមានអង្គភាពចាត់ការនៃការងារ។ វាបានរាយការណ៍ដែលមានអង្គភាពចាត់ការនៃការងារ។

- MOVE ឃ្មាយពើការណានៃឈរសែនពាំង ទៅក្នុងភាព
 - PROPEL ឃ្មាយពើការណាយកាយវត្ថុដីរក្សាប់របស់ខ្លួន
 - INGEST ឃ្មាយពើការណានៃម៉ោងឯកសារឯកសារ
 - EXPEL ឃ្មាយពើការណានៃម៉ោងឯកសារឯកសារ
 - GRASP ឃ្មាយពើការណានៃចំណាំ ឬ ឬវត្ថុ
 - ACT សំលែករាជ្យប្រព័ន្ធសម្រាប់
 - ATRANS ឃ្មាយពើការណាយការណានៃឈរសែនពាំង
 - PTRANS ឃ្មាយពើការណាយការណានៃឈរសែនពាំង
 - ACT សំលែករាជ្យប្រព័ន្ធសម្រាប់
 - SPEAK ឃ្មាយពើការណានៃឈរសែនពាំង
 - ATTEND ឃ្មាយពើការណានៃឈរសែនពាំង
 - ACT សំលែករាជ្យប្រព័ន្ធសម្រាប់ (Mental)

- MTRANS ធមាយពេក្តីរាជការនទ្ទិបាយ
- BUILD ធមាយពេក្តីរាជការព័ត៌មិនិត្យ

3.3.4 ភាល់តាំង

ໃນរាជការនេះទៅទំនើង CD នឹងសាមាតលបុរាណភាល់តាំង នៅ ខេត្តភាល់តាំង ឯមិតាយលខូរ គំព់គំង:

(null)	ប័ណ្ណឃុំ
p	ធមិតិ
f	ធមិនាគិត
/	បច្ចិនសេដ
ts	រាងប្បៀនសម្រាប់
tf	សិនសុណរាងប្បៀនសម្រាប់
c	ក្រើនឈើ
k	តំរីនឈើ
?	តាមរាង

3.3.5 វិធីរាជការនេះបច្ចុប្បន្នបច្ចុប្បន្ន ដោយរាងរឿងរាងរាង

1. ឈរកិច្ចការបច្ចុប្បន្នបច្ចុប្បន្ន ខ្លួន ACTOR, TIME, ACTION, OBJECTS និង DIRECTION.
2. ឱ្យតាមរីនីខ្លះរាងរាង ACTION នឹងខ្លួន រាងយើងរាង (ឯកសារ) និងឱ្យរាងរាង ឱ្យមិនមែនរាងរាង ឬតាមរីនីខ្លះរាងរាង PROPEL
3. ឱ្យតាមរីនីខ្លះរាងរាង រាងយើងរាង និង រាងយើងរាង CD ខ្លួនមាន
ពិរិយាយការងារ CD ខ្លួន:

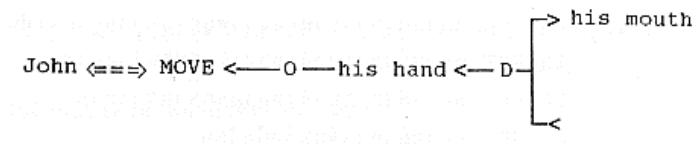
1. ឈរកិច្ចការបច្ចុប្បន្នបច្ចុប្បន្ន

ACTOR: John
TIME: present
ACTION: moves
OBJECT: his hand
DIRECTION: FROM: ចំណុច (រីនីខ្លះរាងរាងបំផុតរាងរាង)
TO: his mouth

2. ឱ្យតាមរីនីខ្លះរាងរាង ACTION នឹង

ໃນທີນີ action ຕື moves ເຊິ່ງຄືກັບໃນ CD ຕື MOVE

3. ພິຈາລະນາຮູບແບບຂອງກົດ ແລະ ການຂຽນຮູບ CD ຂຶ້ນມາ



ໂດຍໃຊ້ກົດທີ 1 PP \Leftrightarrow ACT

ໂດຍໃຊ້ກົດທີ 3 ACT $\leftarrow^0 PP$



ຂໍ້ຕີຂອງການສະແດງຄວາມຮູ້ແບບເພິ່ງພາດ້ານມະໂນພາບຄືການຫຼຸດຂັ້ນຕອນການຫາຂໍ້ສະຫຼຸບ.

ເພາະໃນການສະແດງຄວາມຮູ້ແບບ CD ຈະເປັນໂຄງສ້າງທີ່ກຳນົດຈາກການຫາຂໍ້ສະຫຼຸບໄວ້ລ່ວງໜ້າ ແລ້ວ. ແຕ່ຈຸດດ້ອຍແມ່ນວິທີການສະແດງຄວາມຮູ້ແບບນີ້ຂັບຂ້ອນໝາຍ ແລະ ການກຳນົດຄຳສັ່ງ ມາດຕະຖານໃຫ້ຄົບຖ້ວນຮັບການສະແດງຄວາມຮູ້ທີ່ຫຼາກໝາຍເຮັດໄດ້ຢາກໝາຍ.

ບົດທີ 4: ການຫາຂໍ້ສະຫຼຸບພາຍໄຕຄວາມບໍ່ແນ່ນອນ (Inference under Uncertainty)

ການຮຽນຮັບ ແລະ ການຫາເຫດຜົນໃນປັນຍາປະດິດຈະຕ້ອງພົບກັບຄວາມຮູ້ຕ່າງໆຢ່າງຫຼວງໝາຍ, ທີ່ເປັນຄວາມຮູ້ແບບລົງບ່າຍໃຈຄວາມສົມບຸນ ແລະ ຄວາມຮູ້ທີ່ມີຄວາມຊັບຂອນຈົນຕ້ອງອາໄສປະສົບການເພື່ອສ້າງຄວາມເຂົ້າໃຈສໍາລັບຄວາມຮູ້ ແລະ ຂຶ້ມູນທີ່ມີຄວາມຈະແຈ້ງຈະສະດວກໃນການນຳໄປໃຊ້.

4.1 Uncertainty

ຄວາມຮູ້ທີ່ຜ່ານມາສ່ວນຫຼາຍເປັນຄວາມຮູ້ທີ່ມີໃຈຄວາມສົມບູນ ແລະ ເນື້ອໃນຈະແຈ້ງສາມາດຫາຂໍ້ສະຫຼຸບໄດ້ຈາກການຄົ້ນຫາຄ່າຄວາມຈິງ, ແຕ່ຕົວຈິງແລ້ວປະໂຫຍກ ຫຼື ເນື້ອໃນບາງສ່ວນໃນອີກທາງໜຶ່ງເຫັນວ່າຂາດຄວາມຈະແຈ້ງ ແລະ ມີຄວາມຊໍາຊັອນ ເຮັດໃຫ້ເກີດການສົງໄສ ແລະ ບໍ່ໝັ້ນໃຈ ສິ່ງທີ່ເກີດຂຶ້ນຄົດໆງໍານັ້ນຈຶ່ງເອີ້ນວ່າ: “ຄວາມບໍ່ແນ່ນອນ (Uncertainty)”.

ຄວາມບໍ່ແນ່ນອນ ໝາຍເຖິງສະຖານະການຄວາມລັງເລ, ສົງໃສ, ບໍ່ແນ່ໃຈ ຫຼື ບໍ່ໝັ້ນໃຈ ສົງຜົນ
ໃຫ້ສະຖານະການເກີດຄວາມບໍ່ໝັ້ນຄົງ ແລະ ຜົນຮັບບໍ່ໄດ້ຄວາມແນ່ນອນ ເມື່ອງຈາກຂາດປັດໃຈທີ່ຈະ
ຮັບປະກັນຄວາມຖືກຕ້ອງ ແລະ ສົມບູນ. ຄວາມບໍ່ແນ່ນອນເຫຼົ່ານີ້ອາດເກີດໄດ້ຈາກຫຼາຍສາເຫດຂຶ້ນຢູ່ກັບ
ສະຖານະການຕ່າງໆດັ່ງ: ການລໍາຫຼວດ ແບບໃຊ້ແບບສອບຖາມ ເຊິ່ງຜູ້ຕອບແບບສອບຖາມແຕ່ລະຄົມ
ອາດມີຄວາມເຫັນບໍ່ກົງກັນ, ໃນບາງສ່ວນອາດມີຄວາມເຫັນແຕກຕ່າງຈາກຄົນສ່ວນຫຼາຍ ໂດຍສື່ນເຊິ່ງເຮັດ
ໃຫ້ມີຄ່າຄວາມບໍ່ແນ່ນອນເກີດຂຶ້ນ, ຫັງນີ້ອາດແຕກຕ່າງກັນຕາມກຸ່ມເປົ້າໝາຍທີ່ຕອບແບບສອບຖາມດ້ວຍ.
ຄວາມບໍ່ແນ່ນອນອາດເຮັດໃຫ້ເກີດການຜັນປ່ຽນຂອງຂຶ້ນໝູນເຊິ່ງເກີດຈາກປັດໃຈຫຼາຍຢ່າງ ແຜນ:

4.1.1 ການແກ່ນຄ່າຄວາມບໍ່ແມ່ນອນ

ການແທນຄ່າຄວາມບໍ່ແນ່ນອນສາມາດເຮັດໄດ້ຫຼາຍວິທີ ເຊິ່ງແຕ່ລະວິທີຈະມີການນຳສະເໜີຄ່າຄວາມບໍ່ແນ່ນອນດ້ວຍຂໍ້ມູນຫຼາຍລັກສະນະ ວິທີການພື້ນຖານໃນການແທນຄ່າຄວາມບໍ່ແນ່ນອນມີດັ່ງນີ້

ການແທນຄ່າດ້ວຍຕົວເລກ

ເປັນວິທີການທີ່ງ່າຍແລະສະດວກທີ່ສຸດ ໂດຍກຳນົດຊ່ວງຂອງຕົວເລກເຊັ່ນ 0 ແຫນຄ່າຄວາມບໍ່ແນ່ນອນ ແລະເນື້ອຕົວເລກໝາຍຂຶ້ນສະແດງວ່າຄ່າຄວາມແນ່ນອນໝາຍຂຶ້ນດ້ວຍ

ການແກ່ນຄ້າດ້ວຍກອງປະລາຊົມ

ເປັນວິທີການທີ່ຊ່ວຍລຸດຄວາມບໍ່ໜັ້ນໃຈແລະຄວາມຂັດແຍງທີ່ເກີດຈາກການກຳນົດລະດັບຄວາມບໍ່ແນ່ນອນດ້ວຍຕົວເລກ

ການແທນຄ່າດ້ວຍສັນຍາລັກ

ເປັນການແທນຄ່າຄວາມບໍ່ແນ່ນອນໄດຍການກຳນົດການວັດ ເຊິ່ງຕ້າຍກັບການແທນຄ່າດ້ວຍກຣາບແຕ່ຈະມີການກຳນົດການວັດຢ່າງເປັນລະດັບຢ່າງຈະແຈ້ງ

4.1.2 ການຫາເຫດຜົນແບບ Monotonic

ໃນຂະບວນການທາງປັນຍາປະດິດການໜ້າເຫດຜົນຈະສ້າງດ້ວຍການນຳສະເໜີຄວາມຮູ້ຕ່າງໆ ທີ່ເຖິງຄອມພິວເຕີບໍ່ເຂົ້າໃຈ ຫຼື ສາມາດວິເຄາະ ແລະ ສະຫຼຸບຄວາມໄດ້ຢ່າງຖືກຕອງ. ການໜ້າເຫດຜົນທີ່ມີຄວາມບໍ່ແນ່ນອນຈໍາເປັນຕົງນຳຂະບວນການທາງຕັກກະສາດເຂົ້າມາຊ່ວຍ ເພື່ອໃຫ້ໄດ້ຄວາມຮູ້ທີ່ມີຄວາມຫັດກຸ່ມ ແລະ ວົງວັບຄວາມຈົງທີ່ສຸດ ໃນຂະນະທີ່ເຖິງຄອມພິວເຕີບໍ່ສາມາດເຂົ້າໃຈໄດ້ເຊັນຮັນ. ຕັກກະສາດທີ່ມີຄວາມສຳຄັນໃນການໜ້າເຫດຜົນແບບນີ້ໄດ້ແກ່ Predicate Logic ໂດຍໃຊ້ຫຼັກການທີ່ວ່າ ຄວາມຈົງສາມາດກຳນົດໄດ້ ຫຼື ສ້າງຂຶ້ນມາໃໝ່ຈາກກົດຕ່າງໆ ທີ່ກົງວັບຖານຄວາມຮູ້ (Knowledge Base) ເຊິ່ງເມື່ອກຳນົດຂຶ້ນແລ້ວຄວາມຮູ້ເຖິງນີ້ຈະເປັນຈົງສະເໜີ, ແລະ ຈະບໍ່ຄັດແຍ່ງວັບສິ່ງເກົ່າທີ່ມີຢູ່ແລ້ວ. ເພື່ອເຮັດໃຫ້ຜົນຂອງຂະບວນການທາງຕັກກະສາດ ທີ່ແນ່ນອນການໜ້າເຫດຜົນໃນລັກສະນະນີ້ເອີ້ນວ່າ: “Monotonic”

ການຫາເຫດຜົນດ້ວຍວິທີ Monotonic ຈຳນວນຂອງຄວາມຈິງພືສຸດແລ້ວວ່າຖືກຕ້ອງເມື່ອຮອດໄລຍະໜຶ່ງຈະມີຄ່າຄວາມໝໍາເຊື້ອຖືເພີ່ມຂຶ້ນສະເໝີໂດຍບໍ່ໜຸດລົງ, ແຕ່ກໍມີຂໍຈຳກັດຕ່າງໆບາງອັນເຂົ້າມາເປັນປັດໃຈສົ່ງຜົນໃຫ້ຂໍ້ມູນບໍ່ມີຄວາມສົມບູນພຽງພໍທີ່ຈະໃຊ້ຕັດສິນ ເຊິ່ງເກີດຈາກຄວາມປ່ຽນແປງຂອງຕົວປ່ຽນບາງອັນເຊັ່ນ: ເວລາ. ຂໍຈຳກັດໃນເລື່ອງຄວາມສົມບູນນີ້ທີ່ເຮັດໃຫ້ວິທີການ Monotonic ບໍ່ສາມາດຫາ

ເຫດຜົນຈາກຄວາມຮູ້ໄດ້ ເຊັ່ນ “ ມືນີ້ອາກາດຮັບຄວາມຮູ້ອາໄສການ
ບອກຈາກຄວາມຮູ້ສຶກ ຈຶ່ງຈໍາເປັນຕ້ອງຫາວິທີບອກລະດັບຄວາມຮັບຄວາມຮູ້ອາໄສການ
ບານເຕະແບບບຸກຈະມີໂອກາດຊະນະໝາຍກ່ວາແບບຕັ້ງຮັບ ” ເຊິ່ງຂໍຄວາມນີ້ບໍ່ສາມາດພິສູດຄວາມ
ຖືກຕ້ອງໄດ້ຈະແຈ້ງ ເພາະບໍ່ສາມາດບອກລະດັບຂອງໂອກາດທີ່ຈະຊະນະໄດ້ດ້ວຍແຜນການແນ່ນອນ

ຈາກທີ່ໄດ້ກ່າວມາເຮັດໃຫ້ຕ້ອງອາໄສວິທີການແບບ ເຊິ່ງເປັນຂະບວນການສະແດງຄວາມຮູ້ທີ່
ສາມາດບອກລັກສະນະຄວາມໜ້າເຊື່ອຖືອກມາເປັນຕົວເລັກໄດ້. ຈາກການຄຳນວນຫາຄ່າຄວາມເຊື່ອໝັ້ນ
ໃຫ້ກັບຜົນຮັບທີ່ເກີດຂຶ້ນໃໝ່ໄດ້ ເຊິ່ງຄ່າເຫຼົ່ານີ້ສາມາດປັ່ງນັບປຸງໄດ້ເມື່ອມີຂໍ້ມູນໃໝ່ເພີ່ມຂຶ້ນ ໂດຍລາຍ
ລະອຽດຂອງຄ່າຄວາມແນ່ນອນຈະກ່າວໃນຫົວຂໍ້ຕໍ່ໄປ.

4.1.3 ຄ່າຄວາມແນ່ນອນ

ຄ່າຄວາມແນ່ນອນ (Certainty Factor: CF) ເປັນຄ່າຂອງເວົລັກທີ່ຖືກກຳນົດຂຶ້ນບິນພື້ນຖານ
ຂອງເຫດຜົນ

ຕາຕະລາງທີ 4.1 ສະແດງຄ່າ CF ໃນຊ່ວງຕ່າງໆ ແລະ ຄວາມໝາຍ

Term	Certainty Factor
ບໍ່ແມ່ນຢ່າງແນ່ນອນ (Definitely not)	-1.0
ເກືອບຈະບໍ່ແມ່ນຢ່າງແນ່ນອນ (Almost certainty not)	-0.8
ບໍ່ເປັນໄປໄດ້ (Probably not)	-0.6
ອາດຈະເປັນໄປໄດ້ (Maybe not)	-0.4
ບໍ່ຮູ້ (Unknown)	-0.2 ເຖິງ +0.2
ອາດຈະແມ່ນ (Maybe)	+0.4
ເປັນໄປໄດ້ວ່າຈະແມ່ນ (Probably)	+0.6
ເກືອບຈະແມ່ນຢ່າງແນ່ນອນ (Almost certainty)	+0.8
ແມ່ນຢ່າງແນ່ນອນ (Definitely)	+1.0

CF ຖືກພັດທະນາຂຶ້ນເພື່ອໃຊ້ກັບລະບົບ MYCIN ເຊິ່ງເປັນລະບົບຜູ້ຊ່ວຊານ (Expert System) ສໍາລັບ CF ອາດບໍ່ແມ່ນວິທີການທີ່ເປັນທາງການໃນການກຳນົດລະດັບຄ່າຄວາມບໍ່ແນ່ນອນ, ຕໍ່ການສະໜູບ
ຜົນເປັນໄປຕາມ ການກຳນົດຄ່າ CF ຂອງກົດແຕ່ລະຂຶ້ນໃນລະບົບຈະຖືກກຳນົດຂຶ້ນໂດຍຜູ້ຊ່ວຊານທີ່
ກ່ຽວຂ້ອງກັບສະຖານະການນັ້ນໆ ສໍາລັບຄ່າ CF ທີ່ໄດ້ຈະຂຶ້ນຢູ່ກັບຄວາມຮູ້ ແລະ ປະສົບການຂອງຜູ້ຊ່ວ
ຊານ, ຄ່າ CF ຈະຊ່ວຍບົງບອກເຖິງຄວາມໜ້າເຊື່ອຖືຂອງຂໍສະໜູບໄດ້ຕາມສະຖານະການ ເຊັ່ນ ລົດຄົນ

ໜຶ່ງ ມີເຄື່ອງຈັກ ແລະ ຫຼືໄຟໃນສະພາບປຶກກະຕິ ມີນຳມັນຢູ່ປະລິມານໜຶ່ງ ດັ່ງນັ້ນລົດຄັ້ນດັ່ງກ່າວສະຖານະໃຫ້ງານໄດ້ ໂດຍຜູ້ຊ່ວຊານອາດກຳນົດໃຫ້ຄ່າ CF ເປັນ 0.8 ໃນຂະນະຄູວກັນຫາກມີຜູ້ຊ່ວຊານທີ່ມີປະສົບປະການໝາຍກ່າວອາດກຳນົດຄ່າ CF ເປັນພຽງ 0.6 ຕາມປັດໃຈແວດລ້ອມທີ່ໃຊ້ສະຫຼຸບຜົນເປັນຕົ້ນ [Negnevitsky 2005].

ຄ່າ CF ໃນລະບົບຜູ້ຂໍ່ວຊານຈະນຳມາໃຊ້ຄວບຄຸ້ງກັບຖານຄວາມຮູ້ເຊິ່ງມີຄວາມກຸ່ງວຂ້ອງກັບກົດຕ່າງໆ ທີ່ໃຊ້ໃນການສ້າງຖານຄວາມຮູ້ໂດຍມີໂຄງສ້າງດັ່ງນີ້.

**IF <Evidence (E) >
THEN < Hypothesis (H) { CF }**

ໝາຍວ່າ ທຸກໆຄ່າຂອງ CF ຈະບິງບອກເຖິງຄວາມໜ້າເຊື້ອໃຈໄດ້ວ່າ ເພດການ (E)

จะสั่งผึ้งให้เก็บข้อมูลตาม (H) ในที่สุด ถ้า CF บ่งชี้ว่าจะยุ่งในอุปกรณ์ใดก็ตามได้โดยอิสระตาม 4.1. สำลับที่ดีดีจะเป็น CF จะมีความกู่ว้อยกับฟังชันที่เป็นมาตราตามในงานวัดแยกถ้า Belief และ Disbelief ได้แก่ Measure of Belief (MB(H,E)) และ Measure of Disbelief (MD(H,E)) ฟังชันดังกล่าวจะใช้เป็นตัววัดแยกผลลัพธ์ของความเชื่อที่ ไม่ต่อสัมมุตถามที่กู่ว้อยกับเหตุการณ์ที่ข้างล้วนใจ โดยสามารถหาถ้าได้จากสัมผัสดังนี้:

$$MB(H, E) = \begin{cases} 1 & \text{if } p(H) = 1 \\ \frac{\max[p(H|E), p(H) - p(H)]}{\max[1, 0] - p(H)} & \text{otherwise} \end{cases}$$

$$MD(H, E) = \begin{cases} 1 & \text{if } p(H) = 0 \\ \frac{\min[p(H|E), p(H) - p(H)]}{\min[1.0] - p(H)} & \text{otherwise} \end{cases}$$

ໄຕຍ:

$p(H)$ ໝາຍເຖິງ ຄ່າກະຕວງກ່ອນໜ້າທີ່ສົມມຸດຖານ H ຈະມີຄ່າເປັນຈິງ ໂດຍບໍ່ຢ່າຍໃຫ້ເຫດການໄດ້
 $p(H/E)$ ໝາຍເຖິງ ຄ່າກະຕວງຂອງສົມມຸດຖານ H ທີ່ມີຄ່າຄວາມຈິງເນື້ອຢ່າຍໃຫ້ເຫດການ E

$$CF = \frac{MB(H,E) - MD(H,E)}{1 - \min[MB(H,E), MD(H,E)]}$$

ໂດຍທີ່ຄ່າ CF ທີ່ໄດ້ຈະນຳໄປໃຊ້ໃນການຫາຄ່າຂອງ CF ທີ່ສິນໃຈພາຍໃຕ້ເງື່ອນໄຂຂອງສິນມຸດຖານໄດ້ດັ່ງນີ້: $CF(H,E) = CF(E) \times CF$

ເຊັ່ນ ຕົວຢ່າງ 4.1 ຄວາມຮູ້ໃນການພະຍາກອນອາກາດອາໄສກົດຕໍ່ໄປນີ້

IF ທົ່ວງຟ້າປອດໄປງ

THEN จะพะຍາກອນได้ว่ามีนื้ออาหาดแจ่มใสสุด {CF=0.8 }

ໂດຍທີ່ຄ່າ CF ຂອງຫ້ອງຟ້າປອດໄບງ ຕື: 0.5

ດំណឹង ឈាមទា CF ខេរវាកាលមីនិត្តិមីរឿងខែសាមុជាប់ពីថ្ងៃចុះការបង្ហាញដោយលាស់

$$CF(H,E) = CF(E) \times CF$$

$$CF(H,E) = 0.5 \times .08$$

$$CF(H,E)=0.4$$

ເມື່ອປົງບໍ່ທັງບໍດ້າ CF ທີ່ໄດ້ຈາກຕາຕະລາງທີ່ 4.1 ສາມາດສະຫຼຸບໄດ້ວ່າ “ມີນີ້ອາກາດອາດຈະແຈ່ມໃສ”.

ឃ្លាការា Conjunction

$$CF(H, E1 \wedge E2 \wedge E3, \dots, \wedge En) = min[CF(E1), CF(E2), \dots, CF(En)] \times CF$$

ឃ្លាការណ៍ Disjunction

$$CF(H, E1 \vee E2 \vee E3, \dots, \vee En) = \max[CF(E1), CF(E2), \dots, CF(En)] \times CF$$

ຕົວຢ່າງ 4.2

IF ຂອງຝ້າປອດໄປ່

AND ການພະຍາກອນບອກວ່າອາກາດແຈ່ນໃສ

THEN ຄວນຈະໃສ່ແວ່ນຕາກັນແດດ {CF 0.8}

ໂດຍທີ່ຄ່າ CF ຂອງຂອງຝ້າປອດໄປ່ ຄື: 0.9 ແລະ CF ຂອງການພະຍາກອນວ່າອາກາດແຈ່ນໃສ ຄື: 0.7 ດັ່ງນັ້ນຈະຫາຄ່າ CF ວ່າຄວນຈະໃສ່ແວ່ນກັນແດດພາຍໃຕ້ເຫດການຂຶ້ນໆ.

ໂດຍ E_1 ຄືຂອງຝ້າປອດໄປ່ ແລະ E_2 ຄື ການພະຍາກອນບອກວ່າອາກາດແຈ່ນໃສ

ຈະໄດ້ວ່າ:

$$\begin{aligned} CF(H, E1 \wedge E2) &= \min[CF(E1), CF(E2)] \times CF \\ &= \min[0.9, 0.7] \times 0.8 \\ &= 0.7 \times 0.8 \\ &= 0.56 \end{aligned}$$

ເມື່ອປູງບញ້າຄ່າ CF ກັບຕາຕະລາງ 4.1 ຈຶ່ງສະຫຼຸບໄດ້ວ່າ “ມີນີ້ມີຄວາມເປັນໄປໄດ້ວ່າຄວນຈະໃສ່ແວ່ນກັນແດດ”.

ຈາກທີ່ກ່າວມາຂ້າງເຖິງນັ້ນເປັນການຫາຄ່າ CF ເພື່ອພິຈາລະນາສົມມຸດຖານ ແລະ ເຫດການທີ່ສົນໃຈວ່າມີຄວາມເປັນໄດ້ ແລະ ມີຄ່າຄວາມແນ່ນອນທີ່ຈະເກີດຂຶ້ນຫຼາຍນອຍເຖິ່ງໄດ້ ເຊິ່ງສາມາດນຳໄປປະຍຸກໃຊ້ກັບຄວາມຮູ້ທີ່ສ້າງຂຶ້ນດ້ວຍຫຼັກການໃນລະບົບຜູ້ຊ່ວຂານໄດ້. ຈາກຕົວຢ່າງ ທີ່ຜ່ານມາໄດ້ກຳນົດຄ່າ CF ຖ່າຍ່າຊື້ນມາເພື່ອໃຫ້ເຂົ້າໃຈ ຄວາມຈຶ່ງແລ້ວຄ່າກະຕວງຂອງສົມມຸດຖານ ຫຼື ເຫດການທີ່ສົນໃຈສາມາດຄໍານວນຫາໄດ້ໂດຍໃຊ້ທິດສະດີຄ່າກະຕວງຕ່າງໆ. ໃນບົດນີ້ຈະໄດ້ເວົ້າເຖິງເຕັກນິກການຊອກຫາຄ່າກະຕວງພາຍໃຕ້ຫຼັກການຂອງຄວາມບໍ່ແນ່ນອນ ເຊັ່ນ: ທິດສະດີຂອງ Bayes ແລະ ທິດສະດີຂອງ Demster and Shafer ເປັນຕົ້ນ. ທິດສະດີຕ່າງໆເຖິ່ງມີສະໜັບສະໜູນໃຫ້ການຊອກຫາຄ່າຄວາມແນ່ນອນພາຍໃຕ້ສະຖານະການແວດລ້ອມທີ່ປະກອບດ້ວຍຕົວຢ່າງທີ່ມີຄວາມບໍ່ແນ່ນອນຢູ່ນຳ ແລະ ສາມາດນຳໄປປະຍຸກໃຊ້ໃນລະບົບຕ່າງໆ ຂອງບັນຍາປະດິດໄດ້ຢ່າງມີປະສິດທິພາບ.

ໃນໄລກຄວາມເປັນຈິງ ຄວາມບໍ່ແນ່ນອນເກີດຂຶ້ນໄດ້ສະເໜີ ເນື່ອງຈາກມະນຸດມີຈິດໃຈ ແລະ ສະໝອງທີ່ສຳພັນກັນຫຍາກທີ່ຈະຕັດຂາດຈາກກັນໄດ້, ສົ່ງຜົນໃຫ້ການຕັດສິນໃຈ ຫຼື ການໃຫ້ຂໍ້ມູນອາດເກີດຄວາມຫຼັງເລ ແລະ ຄວາມບໍ່ແນ່ໃຈຂຶ້ນໄດ້. ບາງຄັ້ງອາດຕັດສິນໃຈດ້ວຍສະໝອງທີ່ຂາດຄວາມຮູ້ ຫຼື ຂາດສະຕິ ແລະ ອາດໃຊ້ອາລີມເໜືອເຫດຜົນ ເຮັດໃຫ້ຂໍ້ມູນບາງປະເພດເກີດຄວາມບໍ່ແນ່ນອນໄດ້ສະເໜີ. ການນຳຂໍ້ມູນທີ່ມີຄວາມບໍ່ແນ່ນອນມາໃຊ້ໃນການຈັດຮູບແບບຄວາມຮູ້ເປັນສ່ວນທີ່ຫຼືກລົງໄດ້ຫຍາກ ເຮັດໃຫ້ຈຳເປັນຕົ້ງສຶກສາເລື່ອງຄວາມບໍ່ແນ່ນອນໄວ້. ໃນກໍລະນີທີ່ຄວາມບໍ່ແນ່ນອນເປັນຕົວເລກອາດແກ້ໄຂຫຼື ຫາທີ່ເດສະເມີມາສະໜັບສະໜູນໄດ້ ແຕ່ໃນກໍລະນີທີ່ບໍ່ແນ່ນຮູບແບບຂອງຕົວເລກຄ່າຄວາມບໍ່ແນ່ນອນອາດຕົ້ງໃຊ້ຫຼັກການທີ່ແຕກຕ່າງຈາກເດີມ. ດັ່ງນັ້ນຈຳເປັນຕົ້ງນຳປັດໃຈແວດລ້ອມຂອງຄວາມບໍ່ແນ່ນອນມາພິຈາລະນາອີກດ້ວຍເພື່ອໃຫ້ໄດ້ຜົນຮັບທີ່ໄກ້ຄົງ ແລະ ມີປະສິດທິພາຍໝາຍທີ່ສຸດ.

4.2 ຫິດສະດີ Bayes

ຫິດສະດີ Bayes ເປັນອີກຫຼັກການທີ່ໃຊ້ແກ້ໄຂບັນຫາເລື່ອງຄວາມບໍ່ແນ່ນອນທີ່ອາດເກີດຂຶ້ນຈາກການນຳສະເໜີ ຫຼື ຮຽນຮູ້ຄວາມຮູ້ຕ່າງໆ ໃນລະບົບປັນຍາປະດິດ. ຫິດສະດີ Bayes ເປັນຫິດສະດີທາງສະຖິຕິ ໂດຍນຳຄ່າກະຕວງມາໃຊ້ປະເມີນຄວາມບໍ່ແນ່ນອນເປັນຕົວເລກໄດ້.

ຫິດສະດີ Bayes ກ່ຽວຂ້ອງໂດຍກົງກັບເງື່ອນໄຂຄວາມທີ່ຄາດວ່າຈະເກີດຂຶ້ນດັ່ງນີ້:

ກຳນົດ A ແລະ B ເປັນເຫດການ ແລະຄ່າກະຕວງທີ່ຈະເກີດເຫດການ A ໂດຍມີເງື່ອນໄຂວ່າເຫດການ B ໄດ້ເກີດຂຶ້ນແລ້ວສາມາດຊູ່ມແນນດ້ວຍ “ $P(A/B)$ ” ດັ່ງນັ້ນສາມາດຊູ່ມທີ່ດີການກະຕວງເປັນສົມຜົນໄດ້ດັ່ງນີ້:

$$P(A/B) = \frac{P(B/A)P(A)}{P(B)}$$

ຄ່າກະຕວງຂອງ A ເມື່ອຮູ້ B ຈະຄໍານວນໄດ້ຈາກ ຜົນຖຸນຂອງຄ່າກະຕວງຂອງ B ເມື່ອຮູ້ A ກັບຄ່າກະຕວງຂອງ A ຫັງໝົດຫານດ້ວຍຄ່າກະຕວງຂອງ B ໃນປັນຍາປະດິດການຮຽນຮູ້ຂອງເຄື່ອງຄອມພິວເຕີຈະກ່ຽວຂ້ອງກັບເຫດການ ແລະ ການຕັ້ງສົມມຸດຖານຕ່າງໆ. ຫິດສະດີ Bayes ຈະຊ່ວຍໃນການຊອກຫາສິ່ງທີ່ສົນໃຈຈາກຊຸດຂໍ້ມູນ ຫຼື ຕົວຢ່າງຂໍ້ມູນເພື່ອໃຫ້ໄດ້ຄ່າກະຕວງຂອງສົມມຸດຖານຂອງສິ່ງທີ່ສົນໃຈວ່າມີໂອກາດຈະເກີດຂຶ້ນໄດ້ໝາຍນັ້ນເຖິງໄດ້ໂດຍສາມາດຄໍານວນຕາມຫິດສະດີ Bayes ເຊິ່ງມີສົມຜົນດັ່ງນີ້ [Akerkar, 2005]

$$P(H_i/B) = \frac{P(E/H_i)P(H_i)}{\sum_{n=1}^k P(E/H_n)P(H_n)}$$

P(H_i) = តារាងពេលវេលាដែលមុនពេលថា ស្តីពីការណាមួយ និងការណាមួយទំនាក់ទំនង

$P(H_i/E)$ = ຄ່າກະຕວງກ່ອນໜັ້ນໜ້າທີ່ສົມມຸດຖານ H ທີ່ເປັນຈິງເມື່ອຢູ່ພາຍໃຕ້ເຫດການ E

$P(E/H_i)$ = តារាងទວງໄດៃទីរាយនៃការងារមុនពេលការងារ E ដើម្បីកែតាមការងារ H បៀវជីវិត

k = จำนวนวนที่สิ่งมุตตาณที่เป็นไปได้

ທີ່ດະສະດີ Bayes ຈະນຳມາປະຢູກໃຊ້ກັບປັນຍາປະດິດ ເພື່ອຄໍານວນຫາຕົວເລກຂອງການວິເຄາະ
ຫຼື ການບົ່ງມະຕີສົມມຸດຖານຂອງເຫດການຕ່າງໆ ໂດຍ H ຈະເປັນຕົວກຳນົດຜົນຂອງ ເຫດການ E
ພາຍໃຕ້ເງື່ອນໄຂທີ່ກຳນົດ ແລະ $P(H_i/E)$ ຈະສະແດງເຖິງຄວາມແນ່ນອນຂອງເຫດການໄດ້ອີກດ້ວຍ.
ເຊິ່ງຄວາມສຳພັນດັ່ງກ່າວຊ່ວຍໃນການຕັດສິນໃຈວ່າເຫດການ ຫຼື ຂໍສົມມຸດຖານທີ່ປັນຍາປະດິດພົບພື້ນມີ
ຄວາມໜ້າເຊື້ອໂທິ່ງໝາຍໜ້ອຍເທົ່າໄດ້ ເຊັ່ນ ການບົ່ງມະຕີເຊື້ອພະຍາດຂອງແພດເຊິ່ງອາດໃຊ້ເຕື່ອງຄອມພິວ
ເຕີມາຊ່ວຍໃນການການຕັດສິນໃຈວ່າຜູ້ປ່ວຍມີອາການໄກຄູ່ງກັບພະຍາດ ຫຼື ພະຍາດຊະນິດໄດ້

ລາຍການ	ຊາມໝາຍເລກ 1	ຊາມໝາຍເລກ 2	ລວມ
ໝາກບານສີແດງ	10	20	30
ໝາກບານສີສົ່ມ	30	20	50
ລວມ	40	40	80

ກຳນົດໃຫ້ A ພມາຍເລກຂາມ

B តិ ម្មាកបានសិស្ស

ດំណឹងតារាជពលរដ្ឋបាលទី ១ និង ឈ្មោះជាប្រធានបាល ១ និង ឈ្មោះជាប្រធានបាល ២

$$P(A/B) = \frac{P(B/A)P(A)}{P(B)}$$

ຈາກສັດຂ້າງເທິງຈຳເປັນຕົອງຮັກຄ່າກະຕວງຕໍ່ໄປນີ້:

1. ຄໍາກະຕວງຂອງການຈັບໄດ້ຊາມໝາຍເລກ 1

ຈຳນວນ ຊາມທັງໝົດມີ 2 ໜ່ວຍ ຄ່າກະຕວງທີ່ຈະຈັບໄດ້ 1 ໜ່ວຍຈາກທັງໝົດ 2 ໜ່ວຍຄື

$$P(A) = 1/2 = 0.5$$

2. ຄ່າກະຕວງຂອງການຈັບໄດ້ໝາກບານສີສົມ

- ຊາມໝາຍເລກ 1 ໂອກາດທີ່ຈະຈັບໄດ້ ຕື້ 0.5

ໃນຊາມໝາຍເລກ 1 ມີໝາກບານທັງໝົດ 40 ອັນ ເປັນສີສັນ 30 ອັນ ສະແດງວ່າຄ່າກະຕວງທີ່ຈະຈັບໄດ້ໝາກບານສີສັນ ຕື P(B1)

$$P(B1) = (30/40) \times 0.5 = 0.75 \times 0.5$$

- ຊາມໝາຍເລກ 2 ໂອກາດທີ່ຈະຈັບໄດ້ ຄື 0.5

ໃນຊາມໝາຍເລກ 2 ມີໝາກບານທັງໝົດ 40 ອັນ ເປັນສີສັນ 20 ອັນ ສະແດງວ່າຄ່າກະຕວງທີ່ຈະຈັບໄດ້ໝາກບານສີສັນ ຕື່ $P(B2)$

$$P(B2) = (20/40) \times 0.5 = 0.5 \times 0.5$$

ດំឡើង

$$P(B) = P(B1) + P(B2)$$

$$= (0.75 \times 0.5) + (-0.5 \times 0.5)$$

$$=0.375+0.25$$

$$=0.625$$

ຄ່າກະຕວງຂອງການຈັບໄດ້ມາກົບການສື່ສົ່ມຕີ 0.625

3. ຄ່າກະຕວງຂອງການຈັບໄດ້ມາການສື່ສົ່ມຈາກຊາມໝາຍເລກ 1

ຈາກຂໍ້ 2 ເຮັດໃຫ້ຮູ້ວ່າຄ່າກະຕວງຂອງການຈັບໄດ້ໝາກບານສີສິນຈາກຊາມໝາຍເລກ 1 P(B/A) ຕື່ 0.375 ຈາກຂໍ້ 1 ເຖິງ 3 ສາມາດຄໍານວນຫາຄ່າກະຕວງທີ່ຕ້ອງການໄດ້ດັ່ງນີ້

$$P(A/B) = \frac{0.375 \times 0.5}{0.625}$$

$$= 0.1875/0.625$$

$$= 0.3$$

ຄ່າກະຕວງຂອງການຈັບໄດ້ຊາມໝາຍເລກ 1 ແລະ ເປັນໝາກບານສີສິນ ແມ່ນ 0.3

ຈາກຕົວຢ່າງ 4.3 ການພິຈາລະນາຄ່າກະຕວງທີ່ສິນໃຈຈໍາເປັນຕ້ອງປະກອບດ້ວຍປັດໃຈອົມຂ້າງໝາຍຢ່າງ, ເຊິ່ງຕ້ອງອາໄສທີດສະດີຂອງຄ່າກະຕວງໃນການແກ້ໄຂບັນຫາ. ຈາກຕົວຢ່າງເປັນພຽງການປະຢຸກໃຊ້ໃນລະດັບພື້ນຖານເທົ່ານັ້ນນອກຈາກເຕັກນິກດັ່ງກ່າວແລ້ວ ຍັງມີເຕັກນິກອື່ນໆ ທີ່ໃຊ້ຫຼັກການຄ້າຍຄືກັນແມ່ນ **Full Joint Distribution**

Full Joint Distribution ຈາກຕົວຢ່າງຂ້າງເທິງ ຈະເຫັນໄດ້ວ່າທີດສະດີ ສາມາດຫາຄ່າກະຕວງຂອງສິນມຸດຖານພາຍໃຕ້ສະຖານະການຕ່າງໆໄດ້ ເຊິ່ງຕ່າງກໍເປັນເຫດຜົນຊອດຄອງກັນ, ເນື່ອງຈາກການທີ່ເກີດຜົນຮັບໄດ້ ຈໍາເປັນຕ້ອງມີສາເຫດທີ່ເປັນຕົ້ນກໍາເນີດຂອງເຫດການນັ້ນ. ແຕ່ໃນກໍລະນີຜົນໄດ້ຮັບເກີດຈາກສາເຫດ ຫຼື ເຫດການໝາຍກ່າວໜຶ່ງໃນເວລາດັ່ງກ່າວກັນ ຈໍາເປັນຕ້ອງລວບລວມເຫດການທີ່ໄກຄູງກັນເພື່ອໃຫ້ສາມາດຊອກຫາຄ່າກະຕວງຂອງສິນມຸດຖານໄດ້ຢ່າງຖືກຕ້ອງ. ໃນການລວມເຫດການອາດຕ້ອງອາໄສວິທີການແຍກບາງຄ່າກະຕວງຂອງແຕ່ລະເຫດການເຊິ່ງຊ່ວຍໃຫ້ເຂົ້າໃຈ ແລະ ຄໍານວນໄດ້ງ່າຍຂຶ້ນວິທີການນີ້ເຫັນວ່າ Full Joint Distribution ເປັນວິທີການແຈກຢາຍຄ່າກະຕວງທີ່ໄດ້ຈາກການສ້າງເກັດ ຫຼື ການຈໍາລອງທີ່ນຳໄປໃຊ້ໃນການຄໍານວນຕໍ່ໄປ.

ການແຈກຢາຍຂໍ້ມູນຈະບັນທຶກລົງໃນຕາຕະລາງຕາມຕົວປົງນທີ່ມີ ຫຼື ທີ່ໃຫ້ຄວາມສິນໃຈດັ່ງຕາຕະລາງລຸ່ມນີ້:

ອາການທີ່ກວດພືບ	ເຈັບແຂ້ວ		ປໍເຈັບແຂ້ວ	
	ມີຫົນປຸນ	ບໍ່ມີຫົນປຸນ	ມີຫົນປຸນ	ບໍ່ມີຫົນປຸນ
ເຫືອກອັກເສບ	0.108	0.012	0.072	0.008
ເຫືອກບໍ່ອັກເສບ	0.016	0.064	0.144	0.576

ຕາຕະລາງ 4.3 ສະແດງຄ່າເຈກາຍຄ່າກະຕວງຂອງອາການພາຍໃນຊ່ອງປາກ [Russell and Norving, 2003]

ຈາກຕາຕະລາງຄ່າກະຕວງຂອງແຕ່ລະເຫດການຈະໃໝ່ເຈກາຍໄວ້ຢ່າງຈະແຈ້ງເນື້ອຕ້ອງການພິຈາລະນາເຫດການທີ່ມີຄວາມສຳພັນກັນຈະເປີ່ງໄດ້ຈາກ ແຖວ ແລະ ຖັນ ເຊິ່ງຈະຊ່ວຍໃຫ້ສາມາດດຳນວນຫາຄ່າກະຕວງຂອງຂໍສົມມຸດຖານທີ່ຢູ່ພາຍໃຕ້ເຫດການໜ້າຍກ່ວາໜຶ່ງຢ່າງໄດ້ດັ່ງຕົວຢ່າງ 4.4

ຕົວຢ່າງ 4.4 ພິຈາລະນາຄ່າກະຕວງຂອງອາການຄົນເຈັບທີ່ພົບທັນຕະແພດດັ່ງຕາຕະລາງທີ່ 4.3

ຈົ່ງຫາຄ່າກະຕວງຂອງຄົນເຈັບທີ່ມີອາການເຫືອກອັກເສບ ຫຼືເຈັບ ແຂ້ວ

ຄ່າກະຕວງຂອງຄົນເຈັບທີ່ມີອາການເຫືອກອັກເສບ ຫຼື ເຈັບແຂ້ວຂອກໄດ້ຈາກຜົນບວກຂອງຄ່າກະຕວງທັງໝົດຂອງຄົນເຈັບທີ່ມີອາການເຫືອກອັກເສບ ແລະ ຜົນບວກຂອງຄ່າກະຕວງຂອງຄົນເຈັບທີ່ອາການເຈັບແຂ້ວ ເນື້ອງຈາກເປັນເຫດການທັງໝົດທີ່ສິນໃຈ, ເຊິ່ງພິຈາລະນາຄ່າກະຕວງທີ່ເຈກາຍໃນຕາຕະລາງຕາມທີ່ສິ່ງທີ່ສິນໃຈຮັດໃຫ້ໄດ້ຜົນຮັບດັ່ງນີ້:

ກຸ່ມເຫືອກອັກເສບ ເຈັບແຂ້ວ ແລະ ມີຫົນປຸນ (0.108)

ກຸ່ມເຫືອກອັກເສບ ເຈັບແຂ້ວ ແຕ່ບໍ່ມີຫົນປຸນ (0.012)

ກຸ່ມເຫືອກອັກເສບ ມີຫົນປຸນ ແຕ່ບໍ່ເຈັບແຂ້ວ (0.072)

ກຸ່ມເຫືອກອັກເສບ ພຽງຢ່າງດູງວ (0.008)

ໃນຂະນະທີ່ກຸ່ມຄົນເຈັບທີ່ເຈັບແຂ້ວ (ໂດຍບໍ່ສິນໃຈວ່າຈະມີອາການອື່ນຫຼືບໍ່) ມີທັງໝົດ 4 ກຸ່ມເຊັ່ນດຽວກັບຄົນເຈັບເຫືອກອັກເສບ ແຕ່ໃນກຸ່ມຄົນເຈັບທີ່ເຈັບແຂ້ວຈະມີ 2 ກຸ່ມທີ່ມີອາການເຫືອກອັກເສບ ເຊິ່ງຊ້າກັບກຸ່ມທີ່ກ່າວໄປແລ້ວໃນເບື້ອງຕົ້ນ.

ດັ່ງນັ້ນ ກຸ່ມຂອງອາການເຈັບແຂ້ວຈຶ່ງເຫືອພຽງ 2 ກຸ່ມຄື: ກຸ່ມເຈັບແຂ້ວມີຫົນປຸນ ແຕ່ເຫືອກບໍ່ອັກເສບ (0.016) ແລະ ກຸ່ມທີ່ເຈັບແຂ້ວພຽງຢ່າງດູງວ (0.064).

ຈຶ່ງສະຫຼຸບໄດ້ວ່າ ຄົນເຈັບທີ່ມີເຫືອກອັກເສບ ຫຼື ເຈັບແຂວ້ວ ໂດຍບໍ່ສິນໃຈອາການອື່ນໆ ມີຫັງໝົດ 6 ກຸມດັ່ງນັ້ນ ຄ່າກະຕວງຂອງຄົນເຈັບທີ່ມີເຫືອກອັກເສບ ຫຼື ເຈັບແຂວ້ວ

$$=0.108+0.012+0.072+0.008+0.016+0.064$$

$$= 0.28$$

ຈາກຕົວຢ່າງທີ່ໄດ້ກ່າວມາຈະເຫັນໄດ້ວ່າການລວມເຫດການແບບ ຈະຊ່ວຍໃຫ້ການຄໍານວນເຮັດໄດ້ຢ່າງວ່ອງໄວຂຶ້ນເພາະໂອກາດທີ່ສົມມຸດຖານຈະເກີດຂຶ້ນພາຍໃຕ້ເຫດການຫຼາຍກ່ວາຂຶ້ງຢ່າງນັ້ນເປັນໄປໄດ້ງ່າຍໂດຍສະເພາະການປະຍຸກໃຊ້ວຽກດ້ານລະບົບບັນຍາປະດິດ.

4.3 ຫິດສະດີ Dempster and Shafer

ຫິດສະດີ Dempster and Shafer ເອີ້ນສັນໆວ່າ “ ຫິດສະດີ D-S ” ເປັນຫິດສະດີທາງຄະນິດສາດທີ່ທີ່ກັບທະນາຂຶ້ນມາເພື່ອໃຊ້ໃນການອະທິບາຍຄ່າຄວາມບໍ່ແນ່ນອນ ເຊິ່ງທຳອິດທີ່ກັບທະນາໂດຍ Arthur Dempster ແລະ ຖືກນໍາມາພັດທະນາຕໍ່ໂດຍ Glenn Shafer ເຊິ່ງໄດ້ນຳສະເໜີການອະທິບາຍຄ່າຄວາມບໍ່ແນ່ນອນດ້ວຍຟັງຊັນພິເສດທີ່ເອີ້ນວ່າ “Belief Function”. ຫິດສະດີ D-S ກ່ຽວຂ້ອງກັບການແກ້ໄຂບັນຫາເລື່ອງຄ່າຄວາມບໍ່ແນ່ນອນທີ່ເກີດຂຶ້ນໃນການຕັ້ງສົມມຸດຖານ ໂດຍຫິດສະດີ D-S ມີຈະກຳນິດຊ່ວງຕົວເລກຂອງຄ່າກະຕວງເພື່ອໃຊ້ໃນການສະໜັບສະໜູນສົມມຸດຖານ.

ໃນຫິດສະດີ D-S ຈະມີຄວາມກ່ຽວຂ້ອງກັບກຸ່ມຂອງເຫດການ (Evidence) ຈຶ່ງຈໍາເປັນຕ້ອງມີຄວາມຮັ້ນຖານໃນເລື່ອງກຸ່ມ (Set) ດ້ວຍໂດຍແທນເຫດການຂຶ້ງໆ ເປັນສະມາຊີກພາຍໃນກຸ່ມ, ເຊິ່ງຈະຊ່ວຍໃຫ້ການພິຈາລະນາເຫດການຫັງໝົດທີ່ສາມາດເກີດຂຶ້ນໄດ້ງ່າຍ, ຄວາມຮັ້ນເບື້ອງຕົ້ນໃນຫິດສະດີກຸ່ມທີ່ຈໍາເປັນຕ້ອງສິກສາມີດັ່ງນີ້:

$$\text{ມີ} \quad A = \{x, y, z\}$$

$$\text{ແລ້ວ Power set ຈະເປັນດັ່ງນີ້} \quad P(A) = \{\emptyset, \{x\}, \{y\}, \{z\}, \{x, y\}, \{x, z\}, \{y, z\}, \{x, y, z\}\}$$

ການພິຈາລະນາໃນຫິດສະດີ D-S ຈະດຳເນີນການທີ່ສະມາຊີກພາຍໃນກຸ່ມ, ເຊິ່ງສະມາຊີກດັ່ງກ່າວອານາຍເຫັງ ຄົນ, ສັດ, ສິ່ງຂອງ ເຫດການຕ່າງໆ ໂດຍກຸ່ມທີ່ບັນຈຸສິ່ງເຫຼົ່ານີ້ໃນຫິດສະດີ D-S ເອີ້ນວ່າ “Environment” ເຊິ່ງຈະຊົງແທນດ້ວຍເຄື່ອງໝາຍ ຖ ດັ່ງນີ້:

$$\Theta = \{ \theta_1, \theta_2, \theta_3, \dots, \theta_n \}$$

ຕົວຢ່າງ 4.5

$$\Theta = \{\text{Car}, \text{Van}, \text{Bike}\}$$

ແກນຄ່າດັ່ງນີ້

ລົດ = C(Car) , ລົດຕູ້ = V(Van) , ແລະ ລົດຈັກ = B(Bike)

ຫາກສະຖານະການທີ່ເຮົາສົນໃຈເປັນດັ່ງນີ້:

ພາຫານະໄດທີ່ມີໜ້າຍກ່ວາ 2 ລົ້ມ ຄຳຕອບຄື $\{ \theta_1, \theta_2 \} = \{ C, V \}$

ພາຫານະໄດທີບັນຫຼຸກຄົນໝາຍກ່ວາ 2 ຄົນ ແລ້ວຜິດກົດໝາຍ ຄຳຕອບຄື { θ_3 } = { B }

Environment ດັ່ງກ່າວ ສາມາດຂຽນກຸ່ມສະແດງເຖິງສະມາຊີກທັງໝົດໄດ້ດັ່ງນີ້:

$$P(\Theta) = \{\phi, \{C\}, \{V\}, \{B\}, \{C\}, \{C,V\}, \{C,B\}, \{V,B\}, \{C,V,B\}\}$$

ເຫດການທີ່ສົນໃຈຈະເປັນເງື່ອນໄຂໃນການຈຳແນວກະສະມາຊຸກທີ່ມີຄວາມກົງວຂ້ອງພາຍໃນກຸ່ມນັ້ນ ອອກມາ ເຊິ່ງຈະເຫື້ອພຽງແຕ່ສະມາຊຸກທີ່ມີຄ່າກະຕວງທີ່ເຮີຍສົນໃຈ. ດັ່ງນັ້ນຈຳນວນສະມາຊຸກພາຍໃນກຸ່ມ ປັບປຸງແປງໄປຕາມຄວາມຕ້ອງການ ຫຼື ຂໍ້ຈຳກັດຂອງເຫດການທີ່ສົນໃຈ. ຢ່າງໃດກຳຕາມສະມາຊຸກທີ່ຖືກຈຳ ແນກອອກມານັ້ນທີ່ເປັນກຸ່ມຢ່ອຍ (Subset) ຂອງ ຖ ນັ້ນເອງ.

ທີ່ດີສະດີທີ່ສໍາຄັນອັນຂອງ D-S ທີ່ຈໍາເປັນຕົວສຶກສາ ແລະ ທຳຄວາມເຂົ້າໃຈຄື: ຮູບແບບຂອງພັງຊັ້ນຕ່າງໆທີ່ມີຄວາມສໍາຄັນຕໍ່ການລະບຸຕໍ່ຂອງເຫດການໂດຍນຳຄ່າກະຕວງມາພິຈາລະນາ ແລະ ຄໍານວນຫາຕາມຮູບແບບຂອງແຕ່ລະພັງຊັ້ນ, ສໍາລັບພັງຊັ້ນທີ່ໄດ້ໃນ D-S ມີດັ່ງນີ້:

4.3.1. Mass Function

ໃນທິດສະດີຂອງ D-S ຈະກ່ຽວຂ້ອງໄດຍກົງກັບຄ່າຄວາມເຊື້ອ ຫຼື ລະດັບຄວາມເຊື້ອ (Degree of Belief) ເຊິ່ງຈະເປັນຕົວຢູ່ບອກຄວາມເປັນໄປໄດ້ຂອງເຫດການທີ່ສາມາດເຕີດຂຶ້ນໄດ້ ແຕ່ຄ່າຄວາມເຊື້ອນັ້ນມີຄວາມຄາດເຕືອນສູງ ຈຶ່ງຈໍາເປັນຕ້ອງມີຝັງຊັນທີ່ມາສະໜັບສະໜູນເຫດການນັ້ນ Mass Function

ເປັນຟັງຂັ້ນທີ່ຄ້າຍກັບການບົງບອກເຖິງປະລິມານຂອງມວນສານຂອງວັດຖຸໃນທາງກາຍະພາບ. ສໍາລັບ D-S ແລ້ວຄ່າສ່ວນໝາຍມັກຈະກຳງວຂ້ອງກັບຄ່າກະຕວງ ເຊິ່ງມີຄວາມຊັບຊັ້ນໝາຍໃນການຄືດຄຳນວນດັ່ງນັ້ນຈຶ່ງໃຊ້ ເປັນຕົວແຫນເພື່ອໃຫ້ພິຈາລະນາຄ່າໄດ້ຢ່າງຈະແຈ້ງ, ຂ່ວຍໃຫ້ການລວມ (Combine) ແລະ ແຍກ (Split) ຄ່າໃນເຫດການເຫຼີ້ນນັ້ນຢ່າຍຂຶ້ນ, ຄືກັນກັບການແບ່ງວັດຖຸໜຶ່ງທີ່ມີຄວາມຊັບຊັ້ນເພື່ອໃຫ້ຢ່າຍຕ່າງໆການວັດແທກຄ່າປະລິມານທີ່ແນ່ນອນຂອງວັດຖຸນັ້ນເອງ Mass Function ເຮັ້ນສັນ່ງ ວ່າ “m-function” .

ສໍາລັບ D-S ເຖິງແມ່ນຈະກຳງວຂ້ອງກັບຄ່າຄວາມເຊື່ອຖື (Belief) ແຕ່ດ້ວຍທີດສະດີຂອງ D-S ແລ້ວບໍ່ສາມາດບັງຄັບໃຫ້ເກີດການຍອມຮັບໜີປະຕິເສດສົມມຸດຖານໄດ້ໂດຍກິງ, ສິ່ງທີ່ຕ້ອງການນຳສະເໜີຄືການວັດແທກປະລິມານຄ່າຂອງຄວາມເຊື່ອຖືພາຍໃຕ້ສະພາບແວດລ້ອມທີ່ຖືກກຳນົດຂຶ້ນໃຫ້ຜູ້ສົນໃຈຕັດສິນວ່າ ເຊື່ອຖື (Belief) ຫຼື ບໍ່ເຊື່ອຖື (Nonbelief) , ເຊິ່ງ ໃນທີ່ນີ້ຈະໝາຍເຖິງຄ່າຂອງຄວາມເຊື່ອຕົວໄດ້ທີ່ບໍ່ເປັນກຸ່ມຍ່ອຍພາຍໃຕ້ກຸ່ມຂອງ ແວດລ້ອມ ທີ່ພິຈາລະນາ ຫຼື ສົນໃຈ. ໃນກໍລະນີທີ່ຄວາມເຊື່ອເກີດຂັດແຍ່ງໝື້ ປະຕິເສດສົມມຸດຖານຈະເວົ້າໄດ້ວ່າສົມມຸດຖານນັ້ນເປັນຄ່າ ຍັງບໍ່ເຊື່ອຖືໄດ້ (Disbelief) ເຊິ່ງບໍ່ໄດ້ໝາຍຄວາມວ່າເປັນຄ່າ ບໍ່ເຊື່ອຖືເລີຍ (Nonbelief) ຫຼື ສົມມຸດຖານນັ້ນບໍ່ເປັນຈິງ. ຄ່າຂອງຄວາມເຊື່ອຖື (Belief) ແລະ ບໍ່ເຊື່ອຖືເລີຍ (Nonbelief) ສະແດງດັ່ງຕົວຢ່າງທີ່ 4.6

ຕົວຢ່າງທີ່ 4.6

ຈາກແວດລ້ອມໃນຕົວຢ່າງທີ່ 4.5

$$\Theta = \{\text{Car}, \text{Van}, \text{Bike}\}$$

ແທນຄ່າດັ່ງນີ້

$$\text{ລົດ} = C(\text{Car}), \text{ລົດຕູ້} = V(\text{Van}), \text{ ແລະ } \text{ລົດຈັກ} = B(\text{Bike})$$

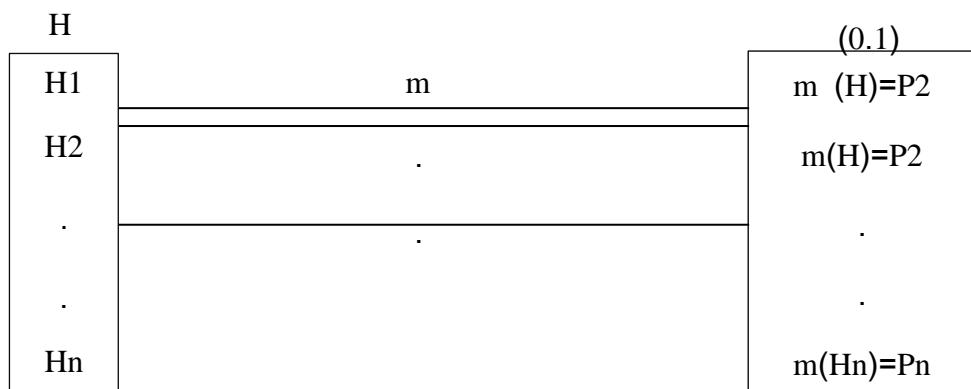
ໜາກສົມມຸດໃຫ້ຄອບຄົວໜຶ່ງມີພາຫະນະຫັ້ງໝົດ 3 ຢ່າງຖື ລົດ, ລົດຕູ້, ແລະ ລົດຈັກ ໂດຍມີຄ່າກະຕວງທີ່ສະມາຊິກໃນເຮືອນຫັງ 3 ຄືນຈະເດີນໜາງດ້ວຍພາຫະນະດັ່ງວັນ ຖີ 0.7 ເມື່ອພິຈາລະນາຈາກພາຫະນະໃນແວດລ້ອມ (Environment) ແລ້ວຈະຮັດໃຫ້ຮູ້ວ່າພາຫະນະທີ່ສາມາດບັນຫຼຸກສະມາຊິກໄປພ້ອມກັນໄດ້ຫັ້ງໝົດ ຖີ ລົດ, ແລະ ລົດຕູ້ ໝາຍເຖິງ $\{C, V\}$ ເຊິ່ງຂຽນເປັນ Mass Function ໄດ້ວ່າ:

ຈາກຄ່າດັ່ງກ່າວຮັດໃຫ້ສາມາດຫາຄ່າ Nonbelief ໄດ້ຈາກຄ່າຂອງ Environment ຫັ້ງໝົດຖື 1 ຕາມທີດສະດີຄວາມໜ້າຈະເປັນ $P(H)+P(\bar{H})=1$ ເມື່ອແທນຄ່າແລ້ວຈະໄດ້ດັ່ງນີ້

$$m_1(\Theta)=1-0.7=0.3$$

ຈາກຕົວຢ່າງທີ່ກ່າວມາຫາງທີ່ດີສະດີໜ້າຈະເປັນຄ່າ 0.7 ຈະໝາຍເຖິງຄວາມເປັນໄປໄດ້ ຫຼື ຄວາມເຊື່ອຂອງເຫດການນັ້ນທີ່ສາມາດເກີດຂຶ້ນໄດ້ ແລະ ຄ່າ 0.3 ຈະໝາຍເຖິງຄວາມບໍ່ເຊື່ອຖື້ນໄອກາດທີ່ເຫດການນັ້ນຈະບໍ່ເກີດຂຶ້ນແຕ່ໃນທີ່ດີ S-D ແມ່ນ 0.7 ຕີ່ Belief ແລະ 0.3 ຕີ່ Nonbelief.

ທີ່ດີສະດີ D-S ຈະອາໄສທີ່ດີສະດີຫຼືນຖານຂອງຄວາມໜ້າຈະເປັນໃນການອະທິບາຍຄ່າຂອງເຫດການທີ່ໜ້າສິນໃຈໄດ້ ໂດຍຈະໃຊ້ຄວບຄຸ່ງກັບ Belief Function ເຊິ່ງຟ້າຊັ້ນຄວາມໜ້າຈະເປັນທີ່ຖືກນຳມາໃຊ້ຄື m-function ເຊິ່ງໝາຍເຖິງຟ້າຊັ້ນຂອງຄວາມໜ້າຈະເປັນຂອງສົມມຸດຖານທີ່ສາມາດອະທິບາຍໄດ້ດ້ວຍຮູບຕໍ່ໄປນີ້:



ຮູບທີ 4.1 ສະແດງກາබຄຸນສົມບັດຂອງ m-function

ຄຸນສົມບັດຂອງ m-function

$$1. \quad 0 \leq m(P) \leq 1$$

$$2. \quad M(\emptyset) = 0$$

$$3. \quad \sum_{P \in H} m(p) = 1$$

ໂດຍທີ່ H ໝາຍເຖິງສົມມຸດຖານ

P ໝາຍເຖິງເຫດການທັງໝົດທີ່ສິນໃຈ ຫຼື ກ່ຽວຂ້ອງກັບສົມມຸດຖານ

ຈາກທີ່ໄດ້ກ່າວໄວ້ໃນເບື້ອງຕົ້ນວ່າທີ່ດີສະດີ D-S ກ່ຽວຂ້ອງກັບຟ້າຊັ້ນຊະນິດໜຶ່ງທີ່ເຮັ້ນວ່າ Belief Function ເຊິ່ງມີຄວາມສໍາພັນກັບ m-function ດັ່ງນັ້ນໃນການສືກສາທີ່ດີສະດີ D-S ຈຳເປັນຕົວເຊົ້າເຖິງນິຍາມຂອງ Belief Function And plausibility Function ດ້ວຍ.

4.3.2 Belief Function

ຄ່າຂອງ Belief Function ໄດ້ຈາກການຮວມກັນຂອງ m-function ຂອງເຫດການຍ່ອຍຫັງໝົດທີ່ໜ້າສິນໃຈຫຼືກ່ຽວຂ້ອງກັບສົມມຸດຖານໂດຍ Belief Function ແຂນແທນດ້ວຍ ‘bel(A)’ ຈາກທີ່ກ່າວມາຫາກຳນົດ A ເປັນເຫດການສົມມຸດຖານໄດ້ ແລະ B ເປັນເຫດການຍ່ອຍໃນ A ແລ້ວຈະໄດ້ສົມຜົນດັ່ງນີ້

$$\text{bel}(A) = \sum_{B \subseteq A} m(B)$$

4.3.3 Plausibility Function

Plausibility Function นี้จะได้จากผนิลรวมของเหตุการณ์อยู่ที่สินใจเชิงมิติความสำ薤ันกับสินมุடาๆ ตามโดยที่ว่าถ้าเป็นค่าของผนิลรวมจากงาน Intersect กันจะหัวใจเหตุการณ์อยู่กับสินมุดๆ ตาม Plausibility function ของแทนด้วย ' $pl(A)$ ' เชิงจะของเป็นสินผนิลได้ดังนี้:

$$p_l(A) = \sum_{B \cap A \neq \emptyset} m(B)$$

สิ่งเดียวที่ความน่าเชื่อถือ (Belief function) และ ความน่าเป็นไปได้ (Plausibility function) มีต่างกัน

$$P(A) = 1 - \text{bel}(A)$$

ໂດຍທີ່ A ຄື Compliment ຂອງ A

▪ Combining Evidence

$$m_1 \oplus m_2(Z) = \sum_{X \cap Y = Z} m_1(X)m_2(Y)$$

จากวิถีดั้งเดิมที่จะเข้ามายังตัวอักษร “ \oplus ” ให้เราสามารถนำสิ่งที่อยู่ในแต่ละเซตมาหาผลรวมกันได้โดยการนำสิ่งที่อยู่ในเซตที่ 1 มาหาผลรวมกับสิ่งที่อยู่ในเซตที่ 2 และนำสิ่งที่อยู่ในเซตที่ 2 มาหาผลรวมกับสิ่งที่อยู่ในเซตที่ 1 ซึ่งจะได้ผลลัพธ์เป็นเซตที่มีสมาชิกที่ไม่ซ้ำกัน นั่นคือ $X \oplus Y = X \cup Y$ หรือ $X \oplus Y = X + Y$ สำหรับเซตที่มีสมาชิกเป็นจำนวนจำกัด เช่น $\{1, 2, 3\} \oplus \{4, 5, 6\} = \{1, 2, 3, 4, 5, 6\}$ แต่สำหรับเซตที่มีสมาชิกเป็นจำนวนอนันต์ เช่น $\mathbb{N} \oplus \mathbb{N} = \mathbb{N}$ หรือ $\mathbb{R} \oplus \mathbb{R} = \mathbb{R}$ ก็จะได้ผลลัพธ์เป็นเซตที่มีสมาชิกที่ไม่ซ้ำกัน แต่จำนวนสมาชิกยังคงเท่าเดิม นั่นคือ $\mathbb{N} \oplus \mathbb{N} = \mathbb{N}$ และ $\mathbb{R} \oplus \mathbb{R} = \mathbb{R}$

ពីរយ៉ាងទី 4.7 តាម Environment និងពីរយ៉ាងទី 4.5

$\Theta = \{\text{Car}, \text{Van}, \text{Bike}\}$

ແກນຄ່າດ້ານນີ້

ລົດ =C(Car), ລົດຕູ້ = V (Van) ແລະ ລົດຈັກ = B (Bike)

ສົມມຸດໃຫ້ຄອບຄົວໜຶ່ງມີພາຫະນະ 3 ຢ່າງຄື ລົດ, ລົດຕູ້ ແລະ ລົດຈັກໂດຍມີຄວາມໜ້າຈະເປັນສະມາຊິກໃນບ້ານທັງ 3 ຄົນຈະເດີນຫາງດ້ວຍລົດຄື 0.9ເຊິ່ງກຳນົດໃຫ້ເຫດການນີ້ເປັນ m_2 ແລະ ຕ້ອງການລວມເຫດການ m_1 (ຈາກຕົວຢ່າງທີ 6,7) ແລະ m_2 ດັ່ງນີ້:

$$m_1(\{C,V\})$$

$$m_1(\Theta)=0,3$$

ແລະ m_1 ຕີ່ $m_2 (\Theta) = 1 - 0.9 = 0.1$

$$M_2 (\{C\}) = 0.9 \quad m_2 (\Theta) = 0.1$$

ຈາກຄໍາເຫດການທັງ m_1 ແລະ m_2 ສາມາດນຳມາຂຽນເປັນຕາຕະລາງເພື່ອສະແດງຄໍາປະເມີນຂອງເຫດການດ້ວຍການ Intersection ດັ່ງນີ້:

Intersection	$m_2 (\{c\}) = 0.9$	$m_2 (\Theta) = 0.1$
$m_1 (\{c,v\}) = 0.7$	$\{c\} = 0.63$	$\{c,v\} = 0.07$
$m_1 (\Theta) = 0.3$	$\{c\} = 0.27$	$(\Theta) = 0.3$

ຕາຕະລາງທີ 4.4 ສະແດງການ Intersection ຂອງ m_1 ແລະ m_2

ຄໍາພາຍໃນຕາຕະລາງມາຈາກສິນການ Dumpster Combination ດັ່ງນີ້

$$T_{11} (\{C\}) = m_1 (\{C,V\}) m_2 (C) = (0.7)(0.9) = 0.63$$

$$T_{21} (\{B\}) = m_1 (\Theta) m_2 (C) = (0.3)(0.9) = 0.27$$

$$T_{12} (\{C,V\}) = m_1 (\{C,V\}) m_2 (\Theta) = (0.7)(0.1) = 0.07$$

$$T_{22} (\{\Theta\}) = m_1 (\Theta) m_2 (\Theta) = (0.3)(0.1) = 0.03$$

ໂດຍ T_{ij} ແກນທີ່ໄດ້ຈາກການ Intersection ໄດ້ຈາກຕາຕະລາງທີ 4.4 ເຊິ່ງ i ຕີ່ແຖວ ແລະ j ຕີ່ ຄໍລົ່າ

ຈາກຕາຕະລາງທີ 4.4 ຈະເຮັດໃຫ້ຮູ້ຄໍາຂອງ Mass Function .ໃໝ່ ຕີ່: m_3 ພາຍໃຕ້ເຫດການທີ່ແຕກຕ່າງກັນດັ່ງນີ້

$$m_3 (\{C\}) = m_1 \oplus m_2 (C) = 0.63 + 0.27 = 0.27$$

$$m_3 (\{C,V\}) = m_1 \oplus m_2 (C,V) = 0.07$$

$$m_3 (\Theta) = m_1 \oplus m_2 (\Theta) = 0.03$$

ຈາກທີ່ກ່າວມາຂ້າງເທິງການລວມເຫດການຈະຊ່ວຍສະໜັບສະໜູນໃນການຄໍານວນຄໍາຂອງ

Belief function ແລະ plausibility function ໃຫ້ຈ່າຍຂຶ້ນໂດຍຄໍາຂອງທັງສອງຟ້າງຊັ້ນຈະບົງບອກເຖິງ ຜົນໃນການປະເມີນຄວາມເຊື້ອຂອງເຫດການເຫຼົ່ານັ້ນໄດ້ ເຊິ່ງຈະພິລາລະນາຈາກຄໍາໄດ້ດັ່ງນີ້:

$0 < bel (A) < 1$ ໝາຍຄວາມວ່າສະໜັບສະໜູນຄໍາຄວາມເຊື້ອ

$0 < pl (A) < 1$ ໝາຍຄວາມວ່າປະຕິເສດຄໍາຄວາມເຊື້ອ

$0 < bel (A) \leq pl (A) < 1$ ໝາຍຄວາມວ່າທັງສະໜັບສະໜູນ ແລະ ປະຕິເສດຄໍາຄວາມເຊື້ອ

ຈາກຫຼັກການທີ່ກ່າວມາຂອງປະຕິທິນ D-S ຈະເຫັນວ່າມີຄວາມກ່ຽວຂ້ອງໂດຍຕົງກັບພື້ນຖານຂອງ ຄວາມໜ້າຈະເປັນ ເພື່ອຫາກວ່າຄໍາຄວາມເຊື້ອຂອງເຫດການຢູ່ໃນລະດັບທີ່ສາມາດສະໜັບສະໜູນຂໍ້ມູນຫຼື ຄວາມຮູ້ດັ່ງກ່າວໄດ້ຫຼືໍຍໍ ເຊິ່ງຈະເຮັດໃຫ້ຮູ້ວ່າມີຄວາມເປັນໄປໄດ້ຫຼາຍໜ່ອຍຊໍາໄດ້ທີ່ຈະເກີດຂຶ້ນ

4.4 Bayesian Network

Bayesian Network เป็นแบบของจำลอง กราฟของความน่าจะเป็น (Probabilistic Graphical Model) ซึ่ง เอ็นโซกัชิมิว่า “Belief Network” ซึ่งเป็นวิธีการนำสังเกตุรุ่มของตัวบ่ง และ ถ้าความน่าจะเป็นที่อิดสະຫຼະตໍ່ກັນ Bayesian Network จะນໍາຄວາມຮູ້ທີ່ໄປທີ່ມີຄວາມສໍາພັນ ຫຼື ກ່ຽວຂ້ອງກັນມານຳສະເໜີເປັນເຊື່ອຂ່າຍທີ່ເຊື່ອມຕໍ່ກັນດ້ວຍຫຼັກຄວາມນ້າຈະເປັນ ແລະ ຂອງຫຼັກຂອງເຫດຜົນໄດ້ຍອາໄສຫຼັກການດັ່ງນີ້ [Kerkira, 2005]

- ຄວາມຮູ້ຕ່າງໆ ໃນໂລກນີ້ລົວປະກອບຂຶ້ນຈາກໜ່ວຍຂອງເຫດການຫຼືໜ່ວຍຂອງອົງປະກອບຕ່າງໆ
- ແບບຈຳລອງຈະນຳສະເໜີເຫດການຕ່າງໆ ທີ່ກ່ຽວຂ້ອງແລະສິ່ງຜົນກະທິບຕໍ່ກັນ
- ເຫດການນີ້ ອາດເປັນອິດສະຫຼະຮັບເຫດການນີ້ ແຕ່ອາດມີຄວາມສໍາພັນກັບບາງເຫດການໄດ້
- ບາງເຫດການອາດມີທິດທາງຄວາມສໍາພັນກັບເຫດການຫຼືໝາງຫາງດູວ ໃນຂະນະທີ່ບາງເຫດການ ອາດມີຄວາມສໍາພັນຫຼາຍກວ່ານີ້ທິດທາງ
- ແຕ່ລະເຫດການສາມາດເຊື່ອມຕໍ່ກັນເປັນແບບຈຳລອງລັກສະນະເຊື່ອຂ່າຍໄດ້

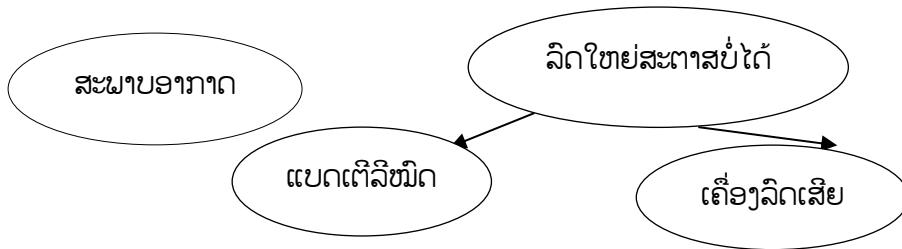
Bayesian Network เป็นแบบຈຳລອງເຊື່ອຂ່າຍທີ່ມີຮູບແບບບໍ່ເປັນວົງວົນ (Cycle) ກ່າວຄືຄວາມສໍາພັນຂອງແຕ່ລະໂນດຈະບໍ່ວິນກັບໄປຫາໂນດເດີມໄດ້ຢັ້ງໃນຂະນະທີ່ມີຄວາມສໍາພັນກັນຕາມທິດທາງທີ່ແບບຈຳລອງນຳສະເໜີເຊື່ອບາງໂນດອາດມີຄວາມເປັນອິດສະຫຼະຈາກໂນດອື່ນໆໄດ້ທີ່ໂນດທັງໝົດໃນແບບຈຳລອງສໍາພັນຫຼືບໍ່ສໍາພັນທັງໝົດກໍໄດ້ການສຶກສາເຖິງຫຼັກການຂອງ Bayesian Network ໄດຍມີຄຸນສົມບັດທີ່ສໍາຄັນດັ່ງນີ້

4.4.1 ໂຄງສ້າງຂອງ Bayesian Network

ຈາກທີ່ກ່າວມາໃນຫົວຂໍ 4.2 ຈະເຫັນໄດ້ວ່າທິດສະດີຂອງ Bayes ຈະມີຄວາມກ່ຽວຂ້ອງກັບເຫດການທີ່ມີຄວາມສໍາພັນກັນດ້ວຍເງື່ອນໄຂຕ່າງໆ ເຮັດໃຫ້ຄ່າຄວາມນ່າຈະຜັນປ່ຽນຕາມຄວາມສໍາພັນຂອງເຫດການນັ້ນໄດ້ກ່າວຄືຄ່າຄວາມນ່າຈະເປັນໃນແຕ່ລະຕົວປ່ຽນອາດເປັນອິດສະຫຼະຕໍ່ກັນດ້ວຍເງື່ອນໄຂຫຼືຄວາມສໍາພັນເຫດການໃນຂະນະດູວກັນອາດມີຕົວປ່ຽນບາງຊະນິດທີ່ບໍ່ເປັນອິດສະຫຼະຕໍ່ກັນ ແລະ ກັນຈະເປັນຕ້ອງມີການເຊື່ອມຕໍ່ກັບເຫດການອື່ນ ເພື່ອສະແດງເຖິງຄວາມສໍາພັນສາມາດອະທິບາຍດ້ວຍການໃຊ້ໂຄງສ້າງຂອງ Bayesian Network ໄດຍມີຄຸນສົມບັດທີ່ສໍາຄັນດັ່ງນີ້:

1. ໂນດທັງໝົດໃນ Bayesian Network ແຕ່ລະໂນດຈະແທນດ້ວຍຕົວປ່ຽນຕ່າງໆທີ່ກ່ຽວຂ້ອງກັບເຫດການຫຼືຂັ້ນມູນທີ່ສົນໃຈ
2. ການເຊື່ອມຕໍ່ລະຫວ່າງໆໂນດດ້ວຍລູກສອນຖ້າລູກສອນອອກຈາກໂນດ X ແລະຊີ້ ຫົວລູກສອນໄປຫາໂນດ Y ຈະເຫັນວ່າ “ ໂນດ X ເປັນໂນດຝໍ່ແມ່ (Parent) ຂອງໂນດ Y ”
3. ແຕ່ລະໂນດ X ຈະມີເງື່ອນໄຂການກະຈາຍຄວາມນ່າຈະເປັນຄື $P(X, I \text{ Parents}(X))$ ເຊິ່ງຈະສິ່ງຜົນຕໍ່ໂນດຝໍ່ແມ່ຂອງແຕ່ລະໂນດ
4. ກຣາບຂອງ Bayesian network ຈະຕັ້ງບໍ່ເຊື່ອມຕໍ່ກັນເປັນວົງຈອນ

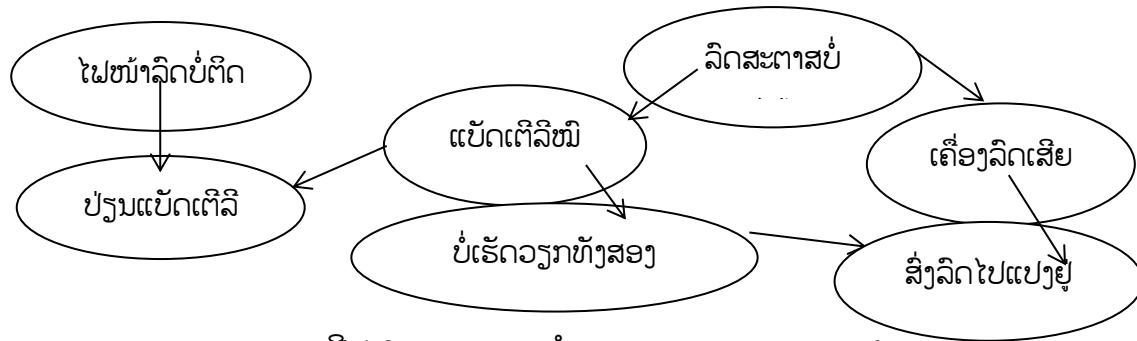
Bayesian Network ຖືກນຳມາໃຊ້ອະທິບາຍຄວາມບໍ່ຂຶ້ນຕໍ່ກັນຢ່າງມີເງື່ອນໄຂ (Condition Independent) ລະຫວ່າງຕົວປຸງນ (Variable) ໂດຍໃນໂຄງສ້າງຂອງ Bayesian Network ມີອາດມີໂນດທີ່ເປັນອິດສະຫະຈາກໂນດອື່ນລວມຢູ່ນຳກໍາໄດ້ພິຈາລະນາຕົວຢ່າງຈາກຮູບທີ 4.2



ຮູບທີ 4.2 ສະແດງໂນດທີ່ເປັນອິດສະຫະພາຍໃນ Bayesian Network

ຈາກຮູບທີ 4.2 ຈະມີຕົວປຸງນຫັງໝົດ 4 ຕົວປຸງນຕີ ສະພາບອາກາດ ລົດສະຕາສບໍ່ໄດ້ ແບບເຕີລີໝົດ ແລະ ເຄື່ອງຈັກລົດເພີ້ເຊິ່ງຕົວປຸງນສະພາບອາກາດນັ້ນຈະບໍ່ມີຄວາມສຳພັນໄດ້ກັບຕົວປຸງນອື່ນເລີຍເຮັດໃຫ້ຕົວປຸງນສະພາບອາກາດເປັນອິດສະຫະບໍ່ຂຶ້ນກັບເງື່ອນໄຂໄດຂອງຕົວປຸງນອື່ນພາຍໃນ Bayesian Network ດັວກັນ ໃນຂະນະທີ່ມີຕົວປຸງນສາມຕົວທີ່ສຳພັນກັນນັ້ນຄືຕົວປຸງນລົດສະຕາສບໍ່ຕິດເປັນໂນດພໍແມ່ (Parent) ຫຼື ເຊັ່ນໄດ້ອີກຢ່າງໜຶ່ງວ່າ “Prior Knowledge” ແລະ ມີຕົວປຸງນແບບເຕີລີກັບເຄື່ອງຈັກລົດເພີ້ຢ່າງໃດຢ່າງໜຶ່ງ, ຢ່າງໄດ້ກຳຕາມຕົວປຸງນຫັງສອງບໍ່ຈໍາເປັນຕ້ອງເປັນສາເຫັດທີ່ເຮັດໃຫ້ເກີດເຫດການລົດສະຕາສບໍ່ຕິດເຫດການດູວແຕ່ອາດເປັນສາເຫັດໃຫ້ເກີດເຫດການອື່ນໄດ້ເຊັ່ນກັນ, ເຊັ່ນວ່າ: ແບບເຕີລີອາດເປັນສາເຫັດທີ່ເຮັດໃຫ້ເກີດໄດ້ຫຼາຍເຫດການ ຫັງລົດສະຕາສບໍ່ຕິດແລະໄຟໝ້າລົດບໍ່ຕິດ (ໃນກໍລະນີໄຟໝ້າລົດບໍ່ຕິດຈະສາມາດສະລຸບໄດ້ຫັນທີ່ວ່າແບບເຕີລີໝົດ) ເປັນຕົ້ນ.

ການສ້າງແບບຈຳລອງ Bayesian Network ມີຈະໃຊ້ເສັ້ນກາບທີ່ມີທິດທາງເພື່ອເຊື່ອມລະຫວ່າງໂນດ ໂດຍແຕ່ລະໂນດຈະມີເງື່ອນໄຈທີ່ສະແດງເຖິງເຫດການຫຼືຄວາມນ່າຈະເປັນຕ່າງໆ ທີ່ອາດເກີດຂຶ້ນກັບຄວາມສຳພັນຮັບໂນດເລີ່ມຕົ້ນ ເຊິ່ງເງື່ອນໄຈທີ່ລະບຸຈະມີຄວາມສຳພັນຮັນຕາມເຫດການຕ່າງໆທີ່ສິນໃຈນັ້ນເອງ ເຊັ່ນ ເມື່ອລົດສະຕາສບໍ່ຕິດ ອາດເກີດຈາກຫຼາຍສາເຫັດ ເຄື່ອງຍິນອາດເກີດຄວາມເສີຍຫາຍແບບເຕີລີໝົດຫຼືອື່ນໆ ໂດຍແຕ່ລະສາເຫັດຈະມີຄວາມຕໍ່ເນື່ອງກັນ (ເງັ້ງຮູບ4.3) ເປັນຕົ້ນ.



ຮູບທີ 4.3 ສະແດງແບບຈຳລອງ Bayesian Network

ຈາກຮູບທີ 4.3 ຈະເຫັນໄດ້ວ່າເຫດການເລີ່ມຕົ້ນຄືລົດສະຕາສບໍຕິດ, ຈະມີເງື່ອນໄຂຫຼືເຫດການທີ່ກ່ຽວຂ້ອງ 2 ໂນດັບ ເຄືອງຈັກລົດເພີ່ມ ແລະ ແບດເຕີລີໝີດ ແລະ ສະແດງວິທີການແກ້ໄຂບັນຫາຈາກຂໍ້ຜິດພາດດັ່ງກ່າວເຊິ່ງຈະມີທິດຫາງແຕກຕ່າງກັນຕາມເຫດການ ຫຼື ເງື່ອນໄຂນັ້ນໂດຍພິຈາລະນາແຕ່ລະສ່ວນວ່າເກີດຈາກເງື່ອນໄຂໃດ, ຫາກແປດເຕີລີໝີດໝັ້ງຍິ່ງຍ່າງດູວກ່າສາມາດແກ້ໄຂໄດ້ໂດຍການປຸ່ງນ, ແຕ່ຖ້າເກີດຈາກເຄືອງຈັກລົດເພີ່ມ ຫຼື ເປັນທັງສອງຍ່າງຈະຕ້ອງສິ່ງລົດໄປແປງທີ່ອຸ່ນ. ນອກຈາກນີ້ເຫດການບາງຢ່າງອາດບໍ່ຍິ່ງບອກທີ່ມາຂອງເຫດທີ່ໜ້າສົນໄດ້ເຊັ່ນ: ໄຟ້ໜ້າລົດບໍ່ຮັດວຽກສະແດງວ່າລະບົບໄຟຟ້າໃນລົດບໍ່ຮັດວຽກຈຶ່ງສະໜູບໄດ້ຫັນທີ່ວ່າແບດເຕີລີໝີດຢ່າງແມ່ນອນ, ໂດຍບໍ່ຈຳເປັນຕ້ອງວິເຄາະເຫດການອື່ນເຊິ່ງຈະສຳພັນກັບເຫດການທີ່ລົດສະຕາສບໍຕິດເປັນຕົ້ນ.

ຈາກຕົວຢ່າງແບບຈຳລອງນີ້ສິ່ງຜົນໃຫ້ເກີດຂະບວນການຕັດສິນໃຈທີ່ແຕກຕ່າງກັນຂຶ້ນ ເນື່ອງຈາກມີປັດໄຈທີ່ສິ່ງຜົນຕໍ່ຜົນຮັບໝາຍກວ່າໜຶ່ງຢ່າງໂດຍສາມາດເລືອກແກ້ໄຂສະຖານະການໄດ້ດັ່ງນີ້:

1. ຫາກລົດສະຕາສບໍໄດ້ ເພາະແປດເຕີລີໝີດ ເຈົ້າຂອງລົດສາມາດແກ້ໄຂສະຖານະການເບື້ອງຕົ້ນໄດ້ໂດຍການປຸ່ງນແບດເຕີລີ
2. ໃນກໍລະນີໄຟ້ໜ້າລົດບໍ່ຕິດສະແດງວ່າລະບົບໄຟຟ້າໃນລົດບໍ່ຮັດວຽກຈຶ່ງບໍ່ຕ້ອງວິເຄາະສາເຫດສະພາບແວດລ້ອມອື່ນກ່ອນ ໂດຍສາມາດແກ້ໄຂບັນຫາໄດ້ຫັນທີ່ດ້ວຍການປຸ່ງນແບດເຕີລີ
3. ຖ້າເຄືອງຈັກມີບັນຫາຮັດໃຫ້ລົດສະຕາສບໍຕິດຈະຕ້ອງສິ່ງລົດໄປແປງຢູ່ອຸ່ນເຫັນນັ້ນ
4. ໃນກໍລະນີທີ່ລົດສະຕາສບໍຕິດເຫດເກີດຈາກເຄືອງຈັກເພີ່ມ ແລະ ແບດເຕີລີໝີດກໍ່ຕ້ອງສິ່ງລົດໄປແປງທີ່ອຸ່ນເຊັ່ນດູວກັນ

ຈະເຫັນໄດ້ວ່າການແກ້ໄຂສະຖານະການສາມາດຮັດໄດ້ໝາຍວິທີຂຶ້ນຢູ່ກັບເງື່ອນໄຂຂອງເຫດນັ້ນດ້ວຍວ່າຈະສິ່ງຜົນຕໍ່ການຕັດສິນໃຈແນວໃດ ໂດຍພິຈາລະນາໄດ້ຈາກທິດຫາງຂອງລູກສອນທີ່ເຊື່ອມຕໍ່ໄປຫາໂນດຕ່າງໆໃນແບບຈຳລອງໂດຍຫຼັກການຂອງ Bayesian Network ນີ້ຈະສາມາດນຳຮູບແບບຂອງທິດຫາງການຕັດສິນໃຈໄປປະຍຸກໃຊ້ກັບບັນຍາປະດິດໄດ້ເປັນຢ່າງດີ. ເນື່ອງຈາກເປັນຮູບແບບການແກ້ໄຂບັນຫາຕາມໜັ້ງກາເຫດຜົນ ແລະ ຄວາມເໝາະສົມຂອງສະຖານະການນັ້ນດ້ວຍ.

4.4.2 ຫິດສະດີຂອງ Bayesian Network

ຈາກທີ່ກ່າວມາຈະເຫັນວ່າ Bayesian Network ມີຄຸນສົມບັດຂອງຄວາມສຳພັນໃນແຕ່ລະໂນດທີ່
ທີ່ເປັນອິດສະຫຼະຈາກກັນ ແລະ ມີເງື່ອນໄຂຕໍ່ກັນເຮັດໃຫ້ໂນດເຫຼົ່ານີ້ຕ້ອງກ່ຽວຂ້ອງກັບຄວາມສຳພັນທີ່ເອີ້ນ
ວ່າ ຄວາມບໍ່ຂຶ້ນຕໍ່ກັນຢ່າງມີເງື່ອນໄຂ (Condition independent) ເຊິ່ງນີ້ຍາຍໄດ້ດັ່ງນີ້

- ទວາມបំផីងតែរក្សាយ៉ាងមិនពីរណ៍ (Condition Independent)

X ບໍ່ຂຶ້ນກັບ Y ຢ່າງມີເງື່ອນໄຂກ່າວຄື ຖ້າຄວາມນໍາຈະເປັນຂອງ X ບໍ່ຂຶ້ນກັບຄ່າຂອງ Y ເມື່ອ
ຮູ້ຄ່າ Z ແລ້ວຈະຂຸ່ນເປັນສົມຜົນດັ່ງນີ້

$$(\forall x_i, y_j, z_k,) P(X=x_i | Y=y_j, Z=z_k) = P(X=x_i | Z=z_k)$$

$$P(X \mid Y, Z) = P(X \mid Z)$$

ຈາກສົມຜິນດັ່ງກ່າວໝາຍເຖິງສໍາລັບ x_i, y_j, z_k ໄດ້ງຄວາມນໍາຈະເປັນທີ່ X ຈະມີຄ່າເປັນ x_i ເມື່ອຮູ້ວ່າ Y ມີຄ່າເປັນ y_j ແລະ Z ມີຄ່າເປັນ z_k ໂດຍມີຄ່າຄວາມນໍາຈະເປັນແມ່ນ X ຈະມີຄ່າເປັນ x_i ເມື່ອຮູ້ວ່າ Z ມີຄ່າເປັນ z_k ເຊິ່ງຈະອື່ນວ່າຄ່າຂອງ X ບໍ່ຂຶ້ນກັບຄ່າຂອງ Y ຢ່າງມີເງື່ອນໄຂເມື່ອຮູ້ຄ່າຂອງ Z ດັ່ງນັ້ນຈຶ່ງສາມາດດັດຕົວປ່ຽນ Y ອອກ ແລະ ຊຽນສົມຜິນໄດ້ເປັນ $P(X | Z)$.

ຄວາມບໍ່ຂຶ້ນຕໍ່ກັນຢ່າງມີເງື່ອນໄຂນີ້ຈະຊ່ວຍໃຫ້ການທາຄວາມນໍາຈະເປັນຂອງຕົວປຸ່ງນີ້ຕ້ອງການງ່າຍຂຶ້ນໂດຍບໍ່ຕ້ອງການສົນໃຈຕົວປຸ່ງນີ້ເຊັ່ນ: ພ້າຮ້ອງ ຈະບໍ່ຂຶ້ນກັບຜົນຕົກສະເໝີໄປຖ້າຮູ້ວ່າເກີດພ້າແມບເຊື່ອໝາຍເຖິງຖ້າເກີດເຫດການພ້າແມບຂຶ້ນສາມາດບອກໄດ້ທັນທີວ່າຈະເກີດສູງພ້າຮ້ອງຕາມໃຈໂດຍບໍ່ຕ້ອງສົນໃຈວ່າຂະນະນັ້ນຜົນຈະຕົກ ຫຼືບໍ່ ເຮັດໃຫ້ການທາຄວາມນໍາຈະເປັນງ່າຍຂຶ້ນໂດຍບໍ່ຕ້ອງສົນໃຈອີກເຫດການໜຶ່ງຈາກທີ່ກ່າວມາສາມາດຂຽນໄດ້ຄື

$$P(\text{Thunder}/\text{Rain, Lighting}) = P(\text{Thunder}/\text{Lighting})$$

ໃນແຕ່ລະໂນດຂອງ Bayesian network ຈຳເປັນຕ້ອງມີຕາຕະລາງຄວາມນໍາຈະເປັນທີ່ເຮື່ອນວ່າ ຕາຕະລາງຄວາມນໍາຈະເປັນແບບມີເງື່ອນໄຂ(Conditional Probability Table) ຫຼື ເຮື່ອນສັ້ນງ່າວ່າ ຕາຕະລາງ CPT ເພື່ອໃຊ້ໃນການປຶກຄົງຄວາມນໍາຈະເປັນຂອງໂນດດ້າງຕົວຢ່າງທີ່ 4.8

ຕົວຢ່າງທີ 4.8

ສະແດງ Bayesian Network ຂອງຄວາມຄວນທີ່ຈະເປັນສະໜາມຫຍໍາຈະປົງກາເນື້ອງຈາກວ່າປັດໄຈທາງດ້ານສະພາບອາກາດ ເຊິ່ງມີຕົວປຸ່ງນັ້ນທັງໝົດ 4 ຕົວປ່ຽນ ໂດຍກໍານົດໄວ້ດັ່ງນີ້:

ມີເມກຫຼາຍ = C (Cloudy) ເປີດ Sprinkler = S (Sprinkler) ຜິນຕົກ = R (Rain) ສະໜາມ
ຫຍ້າປົກ = W (Wet Grass)

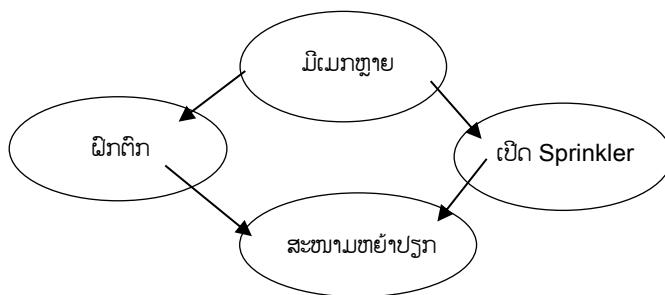
ການເປີດ Sprinkler ກໍສາມາດເປີດໄດ້ຕະຫຼອດເວລາ ເຮັດໃຫ້ຕ້ອງພິຈາລະນາເຖິງປັດໄຈຕົ້ນເຫດທີ່ແທ້ຈິງ ໃນທີ່ນີ້ຕື່ສະພາບອາກາດ.

ເນື່ອງຈາກສະພາບອາກາດມີເມນູາຍມີຄວາມເປັນໄປໄດ້ວ່າທີ່ຈະມີຜົນຕົກໄດ້ຄ່ອນຂ້າງໜູາຍ ເຊິ່ງອາດຈະເປັນສາເຫດຫຼັກທີ່ເຮັດໃຫ້ສະໜາມຫຍ້າປຸງກ ແລະ ໃນບາງໂອກາດອາດຈະເວັດການເປີດ Sprinkler ກໍໄດ້ ແຕ່ການຈະເປີດ Sprinkler ເພື່ອທິດນຳສະໜາມຫຍ້າໃນສະພາບອາກາດຝ້າມີດນັ້ນເປັນໄປໄດ້ຍາກ ຈຶ່ງມີຄ່າຄວາມນ່າຈະເປັນຄ່ອນຂ້າງນ້ອຍ.

❖ ສະແດງຄວາມສໍາພັນຂອງຕົວປົງກ ແລະ ຕາຕາລສົງ CPT ດັ່ງນີ້ບໍ່ທີ່ 4.4

$$P(C) = 0.5$$

C	P(R)
T	0.8
F	0.2



C	P(S)
T	0.1
F	0.5

R	S	P(W)
T	T	0.99
T	F	0.90
F	T	0.90
F	F	0.00

ຮູບທີ່ 4.4 ສະແດງຕົວຢ່າງ Bayesian

Network ແລະ ຕາຕາຕະລາງ

CPT[Russell and Nervig, 2003]

ຮູບທີ່ 4.4 ສະແດງຄວາມນ່າຈະເປັນຂອງແຕ່ລະເຫດການ ໄວ້ໃນຕາຕາຕະລາງ CPT ເຊິ່ງອະທິບາຍໄດ້ວ່າ ໃນສະພາວະບົກກະຕິໂອກາດທີ່ສະພາບອາກາດຈະມີເມນູາຍນັ້ນມີຄວາມນ່າຈະເປັນ ຕື່ $P(C) = 0.5$ ແລະ ເນື່ອສະພາບອາດກາດເປັນລັກສະນະດັ່ງກ່າວ ຈະມີເຫດການທີ່ສາມາດເກີດຂຶ້ນໄດ້ 2 ຢ່າງ ຕື່ ເກີດຜົນຕັກ ແລະ ເປີດ Sprinkler ໂດຍຫັງສອງຢ່າງຈະມີຄວາມນ່າຈະເປັນການເປີດ Sprinkler ຈະມີຄ່ອນຂ້າງຕໍ່ ຕື່ $P(S) = 0.1$ ເນື່ອງຈາກມີໂອກາດສູງທີ່ຜົນຈະຕົກເຮັດໃຫ້ມີຄວາມຈຳເປັນທີ່ຈະເປີດ Sprinkler ລົດລົງນັ້ນເອງ.

ໃນຂະນະທີ່ສະໜາມຫຍ້າປຸງກຈະເກີດຈາກ 2 ເຫດການ ດັ່ງນັ້ນຈຶ່ງຕ້ອງພິຈາລະນາຄວາມອາດຈະເປັນຈາກສອງເຫດການດັ່ງກ່າວເຊິ່ງຜົນທີ່ໄດ້ເປັນດັ່ງຕາຕາຕະລາງຄ່າຄວາມອາດຈະເປັນທີ່ສະແດງໄວ້ໃນຕົວຢ່າງ ເຮັດໃຫ້ຮູບໄດ້ວ່າໂອກາດທີ່ສະໜາມຫຍ້າຈະປູກນັ້ນມີຫຼາຍເຖິງ 0.9 ໃນຫາງກັບກັນໂອກາດທີ່ຫັງສອງເຫດການຈະບໍ່ເກີດຂຶ້ນ ແລະ ເຮັດໃຫ້ສະໜາມຫຍ້າບໍ່ປູກກຳມີເຂັ້ນດູວກັນ.

- ຄວາມອາດຈະເປັນຮ່ວມ (Joint Portability)

ໃນ Bayesian Network ເປັນຕົວປິ່ງແຕ່ລະໄຕຈະມີຄ່າຄວາມອາດຈະເປັນສະເພາະທີ່ອາດເປັນຄວາມອາດຈະເປັນຂອງໂນດໄຕເລີມຕົ້ນ ຫຼື ຄວາມອາດຈະເປັນທີ່ໄດ້ຈາກຄວາມສຳພັນຫຼາຍກວ່າໜຶ່ງໂນດ ໂດຍຄວາມນອາດຈະເປັນທີ່ມາຈາກຕົວປິ່ງຫຼາຍກວ່າໜຶ່ງໄຕເຫັນວ່າ “ຄວາມອາດຈະເປັນຮ່ວມ (Joint Portability)”ດັ່ງນີ້:

$$P(x_1, x_2, \dots, x_n) = \prod_{i=1}^n P(X_i | Parents(X_i))$$

Parents (x_i) ຫຼາຍເຖິງ ໂນດີໍ່ແມ່ໄດ້ຍົກປະກັນ X_i ຈາກສົມຜົນເບື້ອງຕົ້ນສາມາດຫາຄ່າຄວາມອາດຈະເປັນທີ່ X_1, X_2, \dots, X_n ເກີດຂຶ້ນພ້ອມກັນ ໂດຍນຳຄ່າຄວາມອາດຈະເປັນຂອງແຕ່ລະໂນດຄູ່ກັນເຊິ່ງຕ້ອງພິຈາລະນາວ່າແຕ່ລະໂນດຂຶ້ນກົງກັບໂນດີໍ່ແມ່ໂນດໄດ້ວ່າຍ ດັ່ງຕົວຢ່າງທີ່ 4.9

ຕົວຢ່າງທີ່ 4.9

ຈາກຕົວຢ່າງທີ່ 4.8 ຈະສະແດງລາຍລະອຽດຂອງຄວາມອາດຈະເປັນໃນແຕ່ລະຕົວປິ່ງ ຫາກຕົວຢ່າງການທີ່ຈະຮູ້ວ່າຄວາມອາດຈະເປັນທີ່ສະໜາມຫຍ້າຈະປົງກາເພາະຜົນຕົກພຽງຢ່າງດູວ ໂດຍສະພາບອາດກາດໃນຕອນນັ້ນບໍ່ມີເມົກ້າຍທີ່ຈະສົ່ງຜົນເຮັດໃຫ້ຜົນຕົກໄດ້.

ໂດຍພິຈາລະນະຈາກຄ່າຄວາມອາດຈະເປັນໃນຕາຕາຕະລາງ CPT ຂອງແຕ່ລະຕົວປິ່ງ ຕາມຄວາມຕົວຢ່າງການທີ່ກໍານົດຂຶ້ນ

ຄື ສະໜາມຫຍ້າໄດ້ສະພາບອາກາດບໍ່ມີເມົກ້າຍ
ແກ່ນຄ່າລົງໃນສົມຜົນຄວາມອາດຈະເປັນຮ່ວມ ໄດ້ດັ່ງນີ້:

$$\begin{aligned} P(W \wedge S \wedge R \wedge \neg C) &= P(W|S \wedge R)P(S|\neg C)P(R|\neg C)P(\neg C) \\ &= 0.99 \times 0.5 \times 0.2 \times (1-0.5) \\ &= 0.99 \times 0.5 \times 0.2 \times 0.5 \\ &= 0.0495 \end{aligned}$$

ດັ່ງນັ້ນ ຈຶ່ງໄດ້ຄ່າຄວາມອາດຈະເປັນຮ່ວມຂອງສະໜາມຫຍ້າປົງກາໄດ້ອາກາດບໍ່ມີເມົກ້າຍ ຄື 0.0495

ຈາກຕົວຢ່າງທີ່ 4.9 ເມື່ອແກ່ນຄ່າແລ້ວຈະໄດ້ຜົນລັດດັ່ງທີ່ກ່ວາມມາຂ້າງຕົ້ນ ເຊິ່ງຈະເກັນໄດ້ວ່າຕົວປິ່ງນີ້ໄຕອາດບໍ່ຂຶ້ນກັບໄຕໄດ້ເລີຍ ໃນຂະນະທີ່ບາງໄຕຂຶ້ນກົງກັບຕົວປິ່ງອື່ນຫຼາຍກວ່າຕົວປິ່ງໃດໜຶ່ງຫາກພິຈາລະນະຕາມສົມຜົນຫາຄວາມອາດຈະເປັນຮ່ວມແລ້ວ, ຈຳເປັນຕ້ອງມີການຫຼຸດຮູບແບບ ໃຫ້ສອດຄ່ອງກັບຄວາມສຳພັນຂອງແຕ່ລະໂນດນຳ. ໂນດໃດທີ່ບໍ່ຂຶ້ນຕໍ່ກັນກໍ່ບໍ່ຈຳເປັນຕ້ອງຄົງຮູບແບບໄວ້ເພື່ອສະແດງໃຫ້ເກັນຄວາມສຳພັນຂອງແຕ່ລະໂນດໄດ້ຢ່າງຈະແຈ້ງ. ຈາກທີ່ກ່າວວ່າການຫາຄ່າຄວາມອາດຈະເປັນຮ່ວມຈະຕ້ອງຮູ້ຄ່າຂອງຄວາມອາດຈະເປັນຂອງທຸກໂນດ ແຕ່ໃນກໍລະນີທີ່ຮູ້ຄ່າຄວາມອາດຈະເປັນພຽງບາງໂນດຈຳເປັນຕ້ອງໃຊ້ຕັກນິກອື່ນເຂົ້າມາຊ່ວຍນັ້ນແມ່ນ ເຕັກນິກການອະນຸມານທີ່ໃຊ້ສຳລັບ Bayesian Network ເຊິ່ງຈະກ່າວລາຍລະອຽດໃນຫົວຂໍ້ຕໍ່ໄປ.

4.4.3 ການອະນຸມານຂອງ Bayesian Network

ການອານຸມານຂອງ Bayesian Network ຈະໃຊ້ໃນການຫາຄ່າຄວາມອາດຈະເປັນຂອງຕົວປິ່ງຫຼື
ໂນດທີ່ເຮົາສົນໃຈ ເພື່ອໃຊ້ໃນການສະໜັບສະໜູນວິທີອື່ນໆ ຫຼືຊ່ວຍໃຫ້ສາມາດຄໍານວນຫາຜົນຮັບໄດ້ຢ່າງ
ສົມບູນ ເນື້ອງຈາກວ່າໃນການຄໍານວນຈຳເປັນຕົວຮູ້ຄ່າຄວາມອາດຈະເປັນຂອງທຸກໂນດ ເຕັກນິກການ
ອານຸມານຂອງ Bayesian Network ມີດັ່ງນີ້

1. ການອານຸມານຈາກເຫດ (Casual Reasoning)

2. ການອ່ານຸມານຈາກຜົນ (Diagnosis Reasoning)

ເປັນການອານຸມານທີ່ນຳຄ່າຄວາມອາດຈະເປັນຂອງຜົນມາເພື່ອຫາຄ່າຄວາມອາດຈະເປັນຂອງເຫດຕາມທີ່ເຮົາສິນໃຈ ເຊິ່ງກົງກັນຂ້າມກັບລົງທຶນທຳທີ່ໃດ ໃນບາງກໍລະນີການຫາຄ່າຕົວປູງທີ່ເປັນເຫດ ຫຼື ໂນດັ່ງແມ່ອາດຈະເຮັດຍາກ ຈຶ່ງຈຳເປັນຕົວອາໄສ ໂນດລູກທີ່ເປັນຜົນເພື່ອສະໜັບສະໜູນໃຫ້ການຄໍ່ານວນໆຢັ້ງຂຶ້ນ.

3. ການອະທິບາຍເພື່ອລົດຄວາມເປັນໄປໄດ້ (Explaining Away)

เป็นภารกิจของมนุษย์ที่จะต้องรับผิดชอบต่อสังคมและโลก ไม่ใช่แค่การดำเนินการตามกฎหมาย แต่เป็นการสร้างความยั่งยืนให้กับโลก ดังนั้น มนุษย์จึงต้องมีความตระหนักรู้และปรับเปลี่ยนพฤติกรรมอย่างต่อเนื่อง เพื่อให้สามารถรักษาโลกไว้ให้ลูกหลานได้ใช้ประโยชน์ในอนาคต

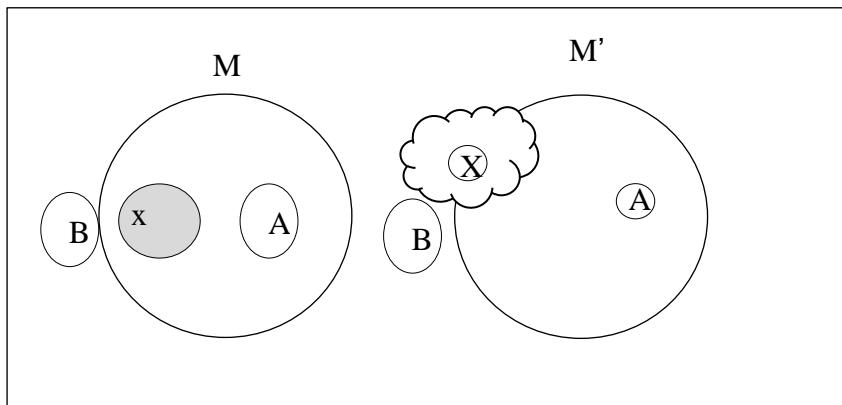
ການອານຸມານແບບຕ່າງໆ ຈະຊ່ວຍໃຫ້ການຫາຄ່າຄວາມອາດຈະເປັນເຮັດໄດ້ຢ່າຍຂຶ້ນ ເນື່ອງຈາກສະຖານະການທີ່ແຕກຕ່າງກັນອາດເຮັດໃຫ້ບໍ່ຮັບຊາບຄ່າຂອງຄວາມອາດຈະເປັນຂອງເຫດການຫັ້ງໝົດດ້ວຍເຫດນີ້ຈຶ່ງພິຈາລະນາເຕັກນິກຂອງການອານຸມານທີ່ເໝາະສົມກັບສິ່ງທີ່ສົນໃຈ ເພື່ອໃຫ້ຄ່າຄວາມອາດຈະເປົ້າທີ່ຕົ້ນອາການໄໝ່ຈາແກ້ຈົດ

4.5 Fuzzy Logic

ເນື່ອງຈາກ Fuzzy Logic ເປັນຫຼັກການທາງຄະນິດສາດທີ່ກ່ຽວຂ້ອງກັບຄວາມບໍ່ແນ່ນອນ ໂດຍທີ່ດີ
ສະດີທາງຄະນິດສາດສ່ວນໝາຍມີຈະນຳມາໃຊ້ກ່ຽວກັບຄວາມອາດຈະເປັນ ແລະ ເຊັດ (Set Theory)
ດ້ວຍເຫດນີ້ Fuzzy Logic ໄດ້ຖືກນຳປະຢຸກໃຊ້ກັບລະບົບປັນຍາປະດິດຕ່າງໆ ເຊິ່ງເນັ້ນໃນການຮຽນອອກ
ແບບລະບົບຄວາມຄົດຂອງສະໜອງມະນຸດ Fuzzy Logic ໄດ້ຖືກພັດທະນາຂຶ້ນ ແລະ ເປັນທີ່ຮູ້ຈັກຄົງທ່າ
ອິດເມື່ອປີ ຕ.ສ 1964-1965 ໂດຍ Lotfi Zadeh ຖືກນຳມາໃຊ້ເພື່ອແກ້ໄຂບັນຫາຂະບວນການຄວາມຄົດ
ຂອງມະນຸດທີ່ຄວບຄຸມລະບົບ ໃຫ້ມີຫຼັກການ ແລະ ມາດຕະການທີ່ຈະແຈ້ງເພື່ອນມຳໄປປະຢຸກໃຊ້ຕໍ່
[Kerkira, 2005].

ແນວຄົດຂອງ Fuzzy Logic ຕີ່ ກຳນົດຄ່າຄວາມອາດເປັນລົງໃນຕົວປ່ຽນ ເພື່ອໃຫ້ຕົວປ່ຽນສາມາດ
ສະແດງຄວາມໝາຍອອກມາໄດ້ຢ່າງຈະແຈ້ງ, ເຊັ່ນ: ຜ້າປູກ ເປັນຄໍາທີ່ລະບຸຄວາມຂຶ້ນຂອງຄ່າທີ່ປູກນຳ້
ແຕ່ກໍບໍ່ສາມາດລະບຸໄດ້ວ່າຜ້າຜົນນີ້ປູກນຳ້ໜ້ອຍ, ໝາຍເທົ່າໄດ ດັ່ງນັ້ນ, ຈຶ່ງມີການລະບຸເພື່ອເພີ່ມນຳ້ໜ້ກ
ກັບຄໍາວ່າ “ປູກ” ໃນທາງກັບກັນກຳຕ້ອງລະບຸຄ່າຂອງຜ້າທີ່ເກືອບແຫ້ງໜີປູກພູງເລັກນ້ອຍດ້ວຍ ເພື່ອ
ສະແດງໃຫ້ເຫັນຄວາມແຕກຕ່າງໆຢ່າງຈະແຈ້ງ ການໃຊ້ຄໍາໃນພາສາບາງຄັ້ງບໍ່ສາມາດທີ່ຈະລະບຸຄວາມ
ແນ່ນອນຂອງສິ່ງທີ່ຕ້ອງການສື່ອກມາໄດ້ຢ່າງຈະແຈ້ງ ດັ່ງນັ້ນ, ຈຶ່ງຈໍາເປັນຕ້ອງນຳ Fuzzy Logic ເຊົ້າ
ມານີສ່ວນຮ່ວມ ຕົວຢ່າງຄໍາທີ່ໃຊ້ໃນການລະບຸນຳ້ໜ້ກ ເຊັ່ນ: ໄກ, ໄກ ແລະ ນ້ອຍເປັນຕົ້ນ.

ແນວຄົດຂອງ Fuzzy Logic ສະແດງດັ່ງຮູບ 4.5



Membership Value of A = 1

Membership Value of B = 0

Membership Value of X = 1

Membership Value of A = 1

Membership Value of B = 0

Membership Value of X: $0 < X < 1$

ຮູບທີ 4.5 ສະແດງແນວຄົດຂອງ Fuzzy Set ແລະ Membership Value [Akerkar, 2005]

ຈາກຮູບທີ 4.5 ເປັນການນີ້ຍາມການກຳນົດຄ່າຂອງ Membership Value ໃຫ້ກັບຂໍ້ມູນທີ່ສິນໃຈ
ເພື່ອໃຫ້ຂໍ້ມູນດັ່ງກ່າວມີຄ່າທີ່ໄກຄົງກັບຄວາມຈິງໝາຍທີ່ສຸດ ໂດຍຈະສະແດງໃຫ້ເຫັນຄ່າຂອງ X ຈະບໍ່ມີ
ພູງຄ່າດູວ ການລະບຸຊ່ວງໃຫ້ກັບຄ່າຂອງ X ຈະຊ່ວຍໃຫ້ການກຳນົດຄ່າດີຂຶ້ນ.

តម្លៃទិន្នន័យ	Membership Value
ធម្មាយ (Very Dry)	0.1
ធម្ម (Dry)	0.3
បានភាព (Average)	0.5
បូណ្ឌ (Wet)	0.7
បូណ្ឌូរ (Very Wet)	0.9

ຕາຕາຕະລາງທີ 4.5 ສະແດງຕົວຢ່າງການກຳນົດຄ່າ Membership Value ຕາມລຳດັບຂຶ້ນທີ່ສິນໃຈ

ຈາກຕາຕາຕະລາງ 4.5 ຈະເຫັນໄດ້ຄໍາຄວາມຊື້ນ ແລະ ຄວາມແຫ້ງແຕ່ລະປະເພດຈະຖືກກຳນົດ
ຄໍາ Membership Value ເພື່ອໃຫ້ເຕີ້ອງຄອມພິວເຕີສາມາດລະບຸສະຖານະຂອງເຫດການທີ່ຕ້ອງປະເຊີນ
ໄດ້ຢ່າງຈະແຈ້ງຂຶ້ນໄດ້ຈາການຄຸມເຕີອທີ່ຕ້ອງປະເຊີນ ຫຼື ຕ້ອງເຈີກບັນຫາທີ່ບໍ່ສາມາດແກ້ໄຂໄດ້ໂດຍສິນ
ຜົນ ຫຼື ຫຼັກການທີ່ມີກົດຈະແຈ້ງຕາຍຕົວ ຫຼັກການ Fuzzy Logic ຈະຊ່ວຍໃຫ້ຂໍ້ມູນມີການຢືນຫຼາຍຂຶ້ນ
ແລະ ຂ່ວຍໃຫ້ຕັດສິນໃຈໄດ້ໄວຂຶ້ນ.

“ສຸພິນເປັນຄົນສູງ”	ເປັນປະໄທຍກທີ່ບໍ່ສາມາດລະບຸເຖິງລະດັບຄວາມສູງທີ່ຈະແຈ້ງໄດ້
“ມື້ນີ້ອາດຮັອນ”	ເປັນປະໄທຍກທີ່ບໍ່ສາມາດລະບຸໄດ້ວ່າອາກາດຮັອນລະດັບໄດ້
“ຫົວບິດສອບເສັງວິຊານີ້ຢາກ”	ເປັນປະໄທຍກທີ່ສະແດງເຖິງຄວາມຍາກຂອງຫົວບິດສອບເສັງ ແຕ່
ບໍ່ຮັກວ່າມັນຫຍາຍເຫຼົ່າໄດ້	

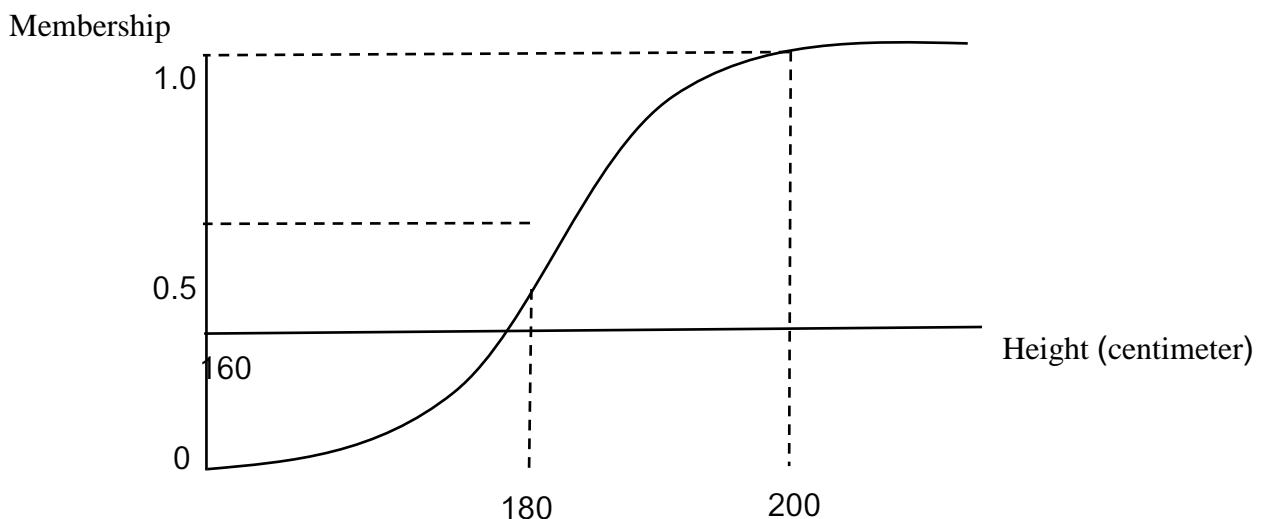
ຈາກປະໂຫຍກທີ່ກ່າວມາຂ້າງເທິງນີ້ ນອກຈາກບໍ່ສາມາດລະບຸຄ່າແນ່ນອນຂອງຄໍາຄຸນນາມໄດ້ແລ້ວ ຍັງບໍ່ຮູ້ແດ່ວ່າປະໂຫຍກທີ່ກ່າວມານີ້ເປັນຈີງ ຫຼື ເຊື້ອໄດ້ຫຼາຍ, ມອຍເທົ່າໃດ ເນື່ອງຈາກບໍ່ມາມີມາດຕະການ ຫຼື ການກຳນົດລະດັບຂອງຄໍາຄຸນນາມເຫັນນັ້ນໄດ້ຢ່າງຈະແຈ້ງ ບາງກຸ່ມຄົນອາດເຫັນດ້ວຍ ຫຼື ຄິດວ່າເປັນ ຈີງ, ແຕ່ໃນບາງກຸ່ມຄົນອາດບໍ່ເຫັນດ້ວຍ ຫຼັງນີ້ເພາະຫັດສະນະຄະຕີ ແລະ ການກຳນົດມາດຕະຖານທີ່

ແຕກຕ່າງກັນ ດັ່ງນັ້ນການກຳນົດຄ່າລະດັບ ຫຼື Membership Value ທີ່ກ່າວໄວ້ໃນຂ້າງເທິງຈຶ່ງມີຄວາມສໍາຄັນຫຼາຍ ເພື່ອໃຫ້ຄໍາຄຸນນາມເຫຼົາມີສະແດງຄວາມໝາຍໄດ້ຖືກຕ້ອງໄດ້ ໂດຍການໃຊ້ຄໍາສະແດງເຖິງນັ້ນກັບອງສິ່ງທີ່ສົນໃຈເຊັ່ນ: ຫຼາຍ, ປານກາງ ແລະ ນ້ອຍ ເປັນຕົ້ນ. ເຊິ່ງຄໍາເຫຼົາມີຈະຊ່ວຍຂະຫຍາຍຄວາມໝາຍທີ່ແນ່ນອນຂອງສິ່ງທີ່ສົນໃຈໄດ້ຫຼາຍຂຶ້ນເຊັ່ນ: ສຸພິນເປັນຄົນສູງຫຼາຍ ຫຼື ຫົວບົດສອບເສັງວິຊານີ້ຍາກທີ່ສຸດ ເປັນຕົ້ນ. ຈາກທີ່ກ່າວມາຈະຊ່ວຍໃຫ້ເຂົ້າໃຈເຖິງນັ້ນກັກ ແລະ ລະດັບຂອງຄໍາຄຸນນາມນີ້ຫຼາຍຂຶ້ນ ໃນ Fuzzy Logic ຈຶ່ງມີການກຳນົດຄ່ານັ້ນກັກ ເພື່ອໃຊ້ແກ້ບັນຫາຄວາມບໍ່ແນ່ນອນທີ່ເກີດຂຶ້ນໃນທຳມະຊາດດັ່ງຕົວຢ່າງຕໍ່ໄປນີ້:

ຕົວຢ່າງທີ 4.10

“ສຸພິນເປັນຄົນສູງ”

ໃນຄວາມເປັນຈຶ່ງສຸພິນສູງ 180 ຊັງຕີແມດ ຈາກປະໂຫຍກດັ່ງກ່າວສະແດງໃຫ້ເຫັນວ່າສຸພິນເປັນຄົນສູງ ແຕ່ບໍ່ສາມາດລະບຸໄດ້ວ່າສູງໃນລະດັບໄດ້, ເຊິ່ງປະໂຫຍກດັ່ງກ່າວອາດຈະເກີດຈາກນຸ່ມມອງຂອງບຸກຄົນທີ່ວ່າໄປ ຫາກມີການກຳນົດຄ່ານັ້ນກັກ ຫຼື Membership Function ໃຫ້ ໂດຍປູປທູບກັບຄ່ານັ້ນກັກຂອງກຸ່ມຄົນທີ່ວ່າໄປ ຈະໄດ້ດັ່ງຮູບທີ 4.6



ຮູບທີ 4.6 ການບັນຫາຄວາມສໍາພັນຂອງ Membership Function ກັບລະດົບຄວາມສູງ

ຈາກຮູບທີ 4.6 ຈະສະແດງໃຫ້ເຫັນຄ່າ Membership Function ຂອງການບັນຫາຄວາມສູງສະເລ່ຍຂອງບຸກຄົນທີ່ວ່າໄປຢູ່ໃນຊ່ວງ 160-200 ຊ.ມ ເຊິ່ງຄ່ານັ້ນກັກຈະເພີ່ມຂຶ້ນຕາມລະດັບຄວາມສູງ ຫາກບຸກຄົນໄດ້ຫາກມີຄວາມສູງນ້ອຍກ່າວ 160 ຊ.ມ ຈະຖືວ່າເປັນຄົນບໍ່ສູງ ໃນຂະນະຄູວກັນກັບບຸກຄົນໄດ້ທີ່ມີຄວາມສູງເກີນ 200 ຊ.ມ ຈະຖືກວ່າເປັນຄົນສູງທັງໝົດ ສຸພິນສູງ 180 ຊ.ມ ຈຶ່ງມີຄ່າ Membership Function ເປັນ 0.5 ຖ້າມີຄວາມສູງ 200 ຊ.ມ ຈະມີຄ່ານັ້ນກັກເປັນ 1.0 ຈາກທີ່ກ່າວມາ ຈຶ່ງຮັດໃຫ້ສາມາດລະບຸຄ່າຂອງຄວາມບໍ່ແນ່ນອນໃນພາສາທຳມະຊາດໃນລະດັບໜຶ່ງ, ເຊິ່ງຊ່ວຍໃຫ້ສາມາດຈຳແນກຄວາມສໍາຄັນຂອງຂໍ້ມູນທີ່ສົນໃຈໄດ້ຢ່າງຈະຈັງຫຼາຍຂຶ້ນ. ໃນນີ້ຂຶ້ນຢູ່ກັບຄວາມເຫັນຂອງບຸກຄົນທີ່ວ່າໄປວ່າຄ່າຄວາມສູງລະດັບໄດ້ທີ່ເຮັ້ນວ່າ “ເປັນຄົນສູງ” ໃນທາງກົງກັນຂ້າມຖ້າກວ່າເປັນການສໍາຫຼວດຈາກກຸ່ມ

ນັກວິລາບານບ່ວງທີ່ມີຄວາມສູງສະເລ່ຍຄ່ອນຂ້າງໜາຍ, ການກຳນົດຊ່ວງລະດັບຄວາມສູງຈະແຕກຕ່າງ
ຈາກເກົ່າຍ່າງຈະແຈ້ງ, ໂດຍກຸ່ມນັກວິລາບານບ່ວງອາດກຳນົດຊ່ວງເປັນ 170-210 ຊ.ມ, ເຊິ່ງຮັດໃຫ້ຄ່າ
ນຳໜັກຂອງຄວາມສູງຂອງສຸພິນຢູ່ໃນລະດັບທີ່ແຕກຕ່າງຈາກເກົ່າຍ່າງແມ່ນອນ ດັ່ງນັ້ນຈຶ່ງຕອງຄຳນິງເຖິງ
ການກຳນົດຄ່ານຳໜັກດ້ວຍ ເພື່ອຄວາມສອດຄ່ອງ ກັບຂໍ້ມູນທີ່ເຮົາສົນໃຈ

ປະໂຫຍດຂອງການນຳ Fuzzy Logic ນຳມາປະຢູກໃຊ້ງານ:

1. ຫຼັກການທີ່ທຳຄວາມເຂົ້າໃຈບໍ່ໄດ້ຍາກ.
2. ໃຊ້ຫຼັກການຄະນິດສາດໃນການສະຫຼຸບຄວາມ ແຮດໃຫ້ມີຄວາມໄກ້ຄົງກັບຄວາມເປັນຈິງ.
3. ເປັນຫຼັກການທີ່ມີຄວາມຢືດຢຸນຕໍ່ຂໍ້ມູນ ແຮດໃຫ້ມີຄວາມຖືກຕ້ອງແມ່ນຍໍາໃນການໜາຜົນຮັບໝາຍ
ຂຶ້ນ.
4. ແກ້ໄຂບັນຫາທີ່ບໍ່ມີຮູບແບບຈະແຈ້ງໄດ້ຢ່າງດີ ໂດຍບໍ່ຕ້ອງອາໄສການຕັດສິນໃຈຕາມກົດເການສະເໜີໄປ.
5. ເປັນການປະສານງານລະຫວ່າປະສິດທິພາບຂອງເຄື່ອງຄອມພິວເຕີ ແລະ ສະໜອງຂອງຜູ້
ຊົງວຊານ.
6. ໃຊ້ພາສາທຳມະຊາດໃນການລະບຸຄ່າ ຫຼື ໃຊ້ຄວາມໝາຍພື້ນຖານຂອງການຕິດຕໍ່ສື່ສານລະຫວ່າງ
ມະນຸດ

ບົດທີ 5 ແນະນຳການຮຽນຮູ້ຂອງເຄື່ອງຈັກ: (Introduction to Machine Learning)

ເຕັກໂນໂລຊີບໍ່ມາປະດິມວ່າກ່ຽວຂ້ອງກັບການຄວບຄຸມເຄື່ອງຈັກ ຫຼື ເຄື່ອງຄອມພິວເຕີໃຫ້ສາມາດເຮັດວຽກໄດ້ຕາມທີ່ຕ້ອງການ, ມີປະສິດທິພາບຄືກັນກັບການຕັດສິນໃຈໄດ້ຍາມນຸດ. ເຄື່ອງຈັກ ຫຼື ເຄື່ອງຄອມພິວເຕີເຫຼົ່ານັ້ນຈໍາເປັນຕ້ອງມີການຮຽນຮູ້ສະຖານະການຕ່າງໆ ເພື່ອໃຫ້ເຂົ້າໃຈ ແລະ ຕັດສິນໃຈແກ້ໄຂບັນຫາຄ້າຍຄືກັບມະນຸດໄດ້ເຫັນຂະບວນການນິ້ວ່າ “ການຮຽນຮູ້ຂອງເຄື່ອງຈັກ” (Machine Learning) ເຊິ່ງເປັນເຕັກໂນໂລຊີທີ່ເນັ້ນໃສ່ການພັດທະນາ ແລະ ອອກແບບຂັ້ນຕອນວິທີທີ່ສະໜັບສະໜູນໃຫ້ເຄື່ອງຈັກຕັດສິນໃຈແກ້ມະນຸດໄດ້ສໍາລັບປົດນີ້ຈະແນະນຳໃຫ້ຮູ້ຈັກກັບການຮຽນຮູ້ຂອງເຄື່ອງຈັກ, ວິທີການສ້າງເຕັກນິກການຮຽນຮູ້ ແລະ ການສ້າງການຕັດສິນໃຈແບບຕົ້ນໄມ້ (Decision Tree) ທີ່ສາມາດນຳໄປໃຊ້ໃນການສ້າງກົດ ຫຼື ຫຼັກການສໍາລັບຄວາມຮູ້.

5.1 ການຮຽນຮູ້ຂອງເຄື່ອງຈຳກ

ໃນປັນຍາປະດິດການຮຽນຮູ້ເປັນປັດໄຈສຳຄັນທີ່ເຮັດໃຫ້ເຄື່ອງຄອມພິວເຕີ ຫຼື ເຄື່ອງຈັກສາມາດຕັດສິນໃຈ, ເຂົ້າໃຈ ແລະ ແກ້ໄຂບັນຫາຕາມສະຖານະການທີ່ເກີດຂຶ້ນໄດ້. ໂດຍທີ່ວໄປເຄື່ອງຄອມພິວເຕີ ຫຼື ເຄື່ອງຈັກຈຳເປັນຕົວອ່າໄສມະນຸດເປັນຜູ້ຄວບຄຸມການເຮັດວຽກທຸກຂັ້ນຕອນ ຕັ້ງແຕ່ປ້ອນຂໍ້ມູນຈົນເຖິງການສະແດງຜົນຮັບ. ດ້ວຍເຫດນີ້ຈຶ່ງມີແນວຕິດທີ່ຈະຫຼຸດບິດບາດການເຮັດວຽກຂອງມະນຸດລົງໄດ້ຢູ່ປັນຍາປະດິດທີ່ມີປະສິດທິພາບໃກ້ຄົງກັບສະໝອງມະນຸດແທນ. ການທີ່ຈະໃຫ້ເຄື່ອງຈັກເຮັດວຽກໄດ້ຢ່າງໜ້າເຊື້ອໂຟ, ຫຼືກຕ້ອງ ແລະ ວ່ອງໄວໄດ້ນັ້ນ ປັນຍາປະດິດທີ່ຢູ່ໃນເຄື່ອງຈັກຕ້ອງມີການຮຽນຮູ້ຂໍ້ມູນຕ່າງໆເພື່ອໃຫ້ສາມາດປະຕິບັດວຽກໄປພ້ອມງ່າກັບການພັດທະນາປະສິດທິພາບຂອງວຽກໃຫ້ສູງຂຶ້ນ. ຄ້າຍຄືກັບການຮຽນຮູ້ຂອງມະນຸດທີ່ສາມາດຈື່ຈຳ ແລະ ສຶກສາວິທີແກ້ໄຂບັນຫາໃນສະຖານະການທີ່ຄ້າຍຄືກັນໄດ້, ເມື່ອມີປະສິບການທຸກຍໍ່ສາມາດແກ້ບັນຫາໄດ້ຢ່າງມີປະສິດທິພາບຫຼາຍຂຶ້ນ.

ເມື່ອເວົ້າເຖິງການຮຽນຮູ້ຂອງເຄື່ອງຈັກໃນດ້ານວິທະຍາສາດຄອມພິວເຕີ ຈະໝາຍເຖິງເຕັກນິກ ຫຼື ຂະບວນການທີ່ໃຊ້ປັບແຕ່ງ, ປັບປຸງອຸປະກອນເຄື່ອງຈັກ ຫຼື ຄອມພິວເຕີໃຫ້ມີຄວາມສາມາດເພີ່ມຂຶ້ນ ມີພິດຕິກຳສະເພາະຕົວທີ່ສະໜັບສະໜູນໃຫ້ອຸປະກອນເຫຼົ່ານີ້ສາມາດຮຽນຮູ້, ເຊົ້າໃຈ ແລະ ສຶກສາຂໍ້ມູນຈາກປະລິບການ ຫຼື ການສະແດງຜົນທີ່ຜ່ານມາໃນອາດີດໄດ້, ເພື່ອໃຊ້ເປັນຂໍ້ມູນສໍາລັບພັດທະນາການໃນການຮັດວຽກຂັ້ນຕໍ່ໄປ. ຈາກທີ່ໄດ້ກ່າວມາເປັນຄວາມໝາຍລວມຂອງການຮຽນຮູ້ຂອງເຄື່ອງຈັກ, ເຊົ້າເປັນວິທີການທີ່ຊ່ວຍໃຫ້ເຄື່ອງຈັກສາມາດຮຽນຮູ້ໄດ້ຄືມະນຸດ. ສໍາລັບປັນຍາປະດິດຄໍາວ່າ ການຮຽນຮູ້ຂອງເຄື່ອງຈັກ ຈະມີຄວາມໝາຍສະເພາະເຈາະຈົງລົງໄປເຖິງການນຳໃຊ້ເຕັກນິກຕ່າງໆ ມາພັດທະນາເຄື່ອງຈັກໃຫ້ມີປະສິດທິພາບໃນການຮຽນຮູ້ສື່ງຕ່າງໆ ບຸງບຄືການສ້າງສະໜອງໃຫ້ກັບເຄື່ອງຈັກເພື່ອໃຊ້ໃນການຮຽນຮູ້ ແລະ ແກ້ໄຂບັນຫາ.

ການຮຽນຮູ້ຂອງເຄື່ອງຈັກໃນປັນຍາປະດິດຈະກ່ຽວຂ້ອງກັບການພັດທະນາຂັ້ນຕອນວິທີ

ຫຼື

ເຕັກນິກຕ່າງໆ ທີ່ຊ່ວຍໃຫ້ເຄື່ອງຈັກ ຫຼື ເຄື່ອງຄອມພົວເຕີມຄວາມສາມາດໃນການຮຽນຮູ້ຂໍ້ມູນທີ່ສິນໃຈໄດ້ເພື່ອຈະນຳໄປໃຊ້ໃນການແກ້ໄຂບັນຫາສະເພາະໜ້າ ທີ່ຕ້ອງພົບພື້ນຖານການຮຽນຮູ້ຂອງມະນຸດທີ່ອາໄສປະລົບການທີ່ໄດ້ຈາກເຫດການຕ່າງໆທີ່ເຕີຍຜ່ານ ມາເກັບສະສົມໄວ້ເພື່ອເປັນຄວາມຮູ້ເຊິ່ງສາມາດນຳມາໃຊ້ໃນການແກ້ໄຂບັນຫາ ແລະ ຕິດຄົ້ນວິທີການທີ່ເໝາະສົມຈົນໄດ້ຜົນຮັບທີ່ຕ້ອງການ.

ການຮຽນຮູ້ໄດຍທີ່ໄປແບ່ງໄດ້ 2 ຊະນິດດັ່ງນີ້:

1. Deductive

ເປັນການຮຽນຮູ້ໄດຍອາໄສຄວາມຮູ້ທີ່ເປັນຈິງຢູ່ແລ້ວ ມີຄວາມຈິງເປັນສາກົນ ເຊິ່ງຫຼຸກຄົນໄດ້ຍອມຮັບໄດຍສາມາດຄາດການໄດ້ວ່າເຫດການດັ່ງກ່າວຈະເກີດຂຶ້ນແນ່ນອນຕາມຮູບແບບຂອງສິ່ງແວດລ້ອມ, ເຊັ່ນ: ໃນກັບໜ່ວຍໜຶ່ງບັນຈຸໝາກບານຢູ່ 2 ໜ່ວຍ ຕີ: ໜ່ວຍສີຟ້າ ແລະ ສີແດງ, ສະແດງວ່າໂອກາດທີ່ຈະຈັບໄດ້ໝາກບານ 1 ໜ່ວຍ ເປັນສີໄດສີໜຶ່ງ ຖີວ່າໄດ້ 1 ໃນ 2 ໜ່ວຍ ເທິງກັບ 50%, ເຊິ່ງເປັນຄ່າກະຕວງທີ່ຫຼຸກຄົນຍອມຮັບກັນ ແລະ ເຊື່ອວ່າຫາກຈັບໝາກບານ 1 ໜ່ວຍ ໂອກາດທີ່ຈະໄດ້ໝາກບານສີໄດ່ ຈະເທິງກັນເປັນຕົ້ນ.

2. Inductive

ເປັນການຮຽນຮູ້ຈາກເຫດການ ຫຼື ສິ່ງທີ່ສິນໃຈ ໄດຍຮູ້ຂໍ້ມູນ ຫຼື ຕ່າຄວາມຈິງພຽງບາງສ່ວນ, ເຊິ່ງຈະເອົາມາໃຊ້ເປັນຂໍ້ມູນໃນການສຶກສາ ແລະ ສ້າງຄວາມເຂົ້າໃຈ ເພື່ອໃຫ້ໄດ້ຄວາມຈິງຂອງຂໍ້ມູນສ່ວນອື່ນໆທັງໝົດ ຈົນເປັນຄວາມຈິງສາກົນທີ່ຄົນສ່ວນໝາຍຍອມຮັບ ເຊັ່ນ: ການຮຽນຮູ້ຂອງພະນັກງານຂາຍ ໄດຍສຶກສາຈາກພິດຕິກຳ, ລັກສະນະຂອງລູກຄ້າ ແລະ ຄວາມສິນໃຈໃນຂະນະນຳສະເໝີສິນຄ້າ ເພື່ອຄົ້ນຫາຄວາມຕ້ອງການທີ່ແທ້ຈິງຂອງລູກຄ້າ, ຊ່ວຍໃຫ້ຮູ້ເຖິງແນວທາງໃນການຂາຍສິນຄ້າແກ່ລູກຄ້າໃຫ້ປະສົບຜົນສໍາເລັດ. ເຊິ່ງເປັນການສຶກສາຈາກຂໍ້ມູນພຽງບາງສ່ວນ ຈົນຮູ້ເຖິງຄວາມຕ້ອງການທີ່ແທ້ຈິງຂອງລູກຄ້າເປັນຕົ້ນ.

ສໍາລັບການຮຽນຮູ້ຂອງເຄື່ອງຈັກຈະເປັນການຮຽນຮູ້ຈາກຂໍ້ມູນເປັນບາງສ່ວນ ກ່ອນຈະດຳເນີນການຫາຄວາມຈິງທີ່ເປັນສາກົນ. ໃນຂະບວນການຂອງການຮຽນຮູ້ຂອງເຄື່ອງຈັກຈະມີການນຳເຕັກນິກ ຫຼື ຂັ້ນຕອນວິທີທີ່ມີຮູບແບບແຕງຕ່າງກັນມາແກ້ບັນຫາໃນລັກສະນະຕ່າງໆ, ເຊິ່ງຂັ້ນຕອນວິທີໃນການຮຽນຮູ້ຂອງເຄື່ອງຈັກມີຢູ່ໝາຍຊະນິດ ສໍາລັບໃນບົດນີ້ຈະກ່າວເຖິງຂັ້ນຕອນວິທີບາງຊະນິດ ໄດ້ແກ່ Supervised Learning ແລະ Unsupervised Learning.

5.1.1 Supervised Learning

ເປັນການຮຽນຮູ້ທີ່ສາມາດນຳສະເໜີ ແລະ ຈຳແນກຂໍ້ມູນພາຍໃນຊຸດຂໍ້ມູນວ່າມີຜົນຮັບທີ່ຖືກ ຫຼື ຜິດໄດ້, ເຊິ່ງໃນຊຸດຂໍ້ມູນດັ່ງກ່າວຈະປະກອບດ້ວຍຂໍ້ມູນເອກະລາດ ແລະ ຂໍ້ມູນທີ່ສິນໃຈ. ຂໍ້ມູນເຫຼົ່ານີ້ຈະ ທີ່ກາເອົາໄປໃຊ້ປະມານຄ່າ ຫຼື ພະຍາກອນຄ່າຂໍ້ມູນ, ໂດຍມີພື້ນຖານການພະຍາກອນຈາກຂໍ້ມູນທັງໝົດໃນ ຊຸດຂໍ້ມູນ. ຕົວຢ່າງເຕັກນິກທີ່ໃຊ້ໃນການຮຽນຮູ້ປະເພດນີ້ຕີ: Decision Tree, Perceptrons ແລະ Backpropagation ເປັນຕົ້ນ.

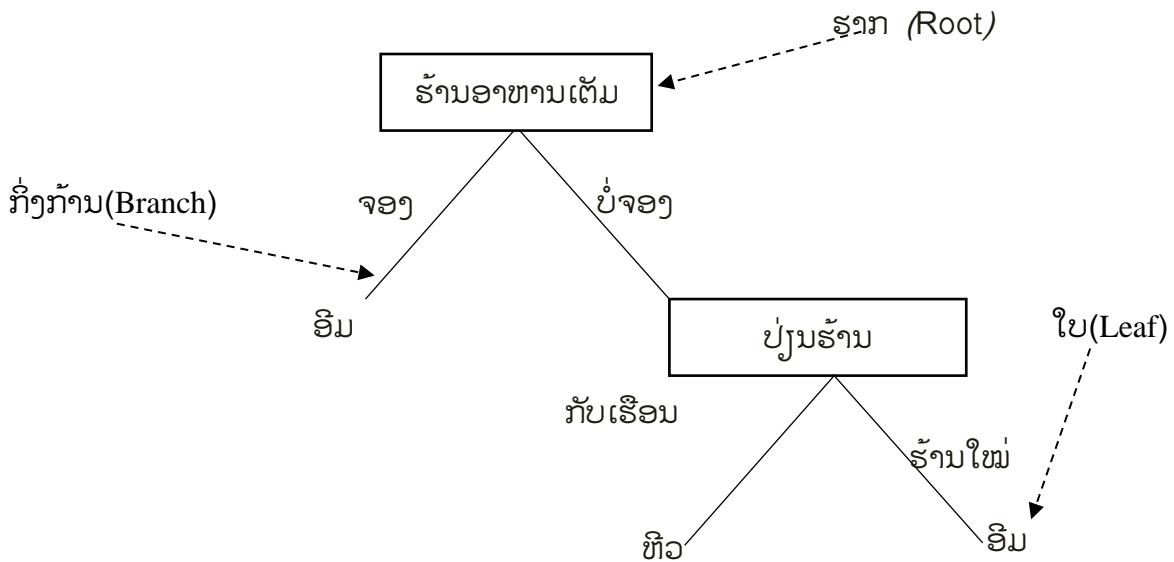
5.1.2 Unsupervised Learning

ເປັນການຮຽນຮູ້ທີ່ບໍ່ມີການກຳນົດຂໍ້ມູນທີ່ສິນໃຈພາຍໃນຊຸດຂໍ້ມູນ, ຈຶ່ງບໍ່ມີການຈຳແນກຂໍ້ມູນວ່າມີຜົນ ຮັບເປັນແນວໄດ້, ແຕ່ຈະເປັນການຮຽນຮູ້ທີ່ກ່ຽວຂ້ອງກັບຄວາມສໍາພັນຂອງຂໍ້ມູນ ເຊິ່ງຈະນຳໄປໃຊ້ໃນ ການຈຳແນກ ແລະ ແຍກຂໍ້ມູນອອກເປັນກຸ່ມ. ຕົວຢ່າງເຕັກນິກທີ່ໃຊ້ໃນການຮຽນຮູ້ປະເພດນີ້ຕີ: Nearest Neighbor Classification ເປັນຕົ້ນ.

5.2 ການຕັດສິນໃຈແບບຕົ້ນໄມ້ (Decision Tree)

Decision Tree ຫຼື Classification Tree ເປັນອີກເຕັກນິກທີ່ຈະຮຽນຮູ້ຂອງເຄື່ອງຈັກທີ່ໃຊ້ ໃນການພັດທະນາການຮຽນຮູ້ຂອງເຄື່ອງຈັກ ຫຼື ເຄື່ອງຄອມພິວເຕີ. ເຕັກນິກນີ້ຈັດເປັນ Supervised Learning, ເຊິ່ງເປັນແບບຈຳລອງທີ່ໃຊ້ສໍາລັບຄາດຄະເນ ຫຼື ທຳມາຍເຫດການທີ່ຈະເກີດຂຶ້ນລວງໜ້າ, ເຊິ່ງເປັນຜົນໄດ້ຮັບທີ່ໄດ້ຈາກການຕັດສິນໃຈ. Decision Tree ເປັນຂັ້ນຕອນໃນການຮຽນຮູ້ ຫຼືບໍ່ຄ່ອຍ ຂັບຊອນປານໄດ້ ໂດຍຈະມີການແຕກແໜ່ງຈາກໄນດຮາກ (Root) ສູ່ໃບ (Leaf) ແລະ ມີກິງກ້ານ (Branch) ແຕກອອກໄປຕາມເງື່ອນໄຂ ຫຼື ຂໍ້ມູນທີ່ໄດ້ຄາດຄະເນໄວ້ຈະເກີດຂຶ້ນ ເພື່ອໃຫ້ຮູ້ເກີງຜົນຮັບ ຂອງແຕ່ລະເຫດການ, ເປັນແບບຈຳລອງທີ່ມີການເຊື່ອມຕໍ່ລະຫວ່າງສິ່ງທີ່ສິນໃຈກັບຜົນສະຫຼຸບທີ່ອາດຈະ ເກີດຂຶ້ນ ຈາກຄ່າຂອງເຫດການຕ່າງໆເຊັ່ນ: ເຈົ້າທີ່ວ່າເຂົ້າບໍ່? ຄໍາຕອບເປັນໄດ້ທັງໝົດ ຫຼື ບໍ່ທີ່ວ່າເປັນຕົ້ນ.

ໂຄງສ້າງຂອງ Decision Tree ຈະປະກອບດ້ວຍໃບເປັນສ່ວນຂອງຂໍ້ມູນທີ່ເຮົາສິນໃຈ, ເຊິ່ງອາດ ເປັນຂໍ້ມູນທີ່ເກີດຂຶ້ນໂດຍສະພາບແວດລ້ອມຕາມສະຖານະການນັ້ນ ຫຼື ເປັນສິ່ງທີ່ກຳນົດຕາມການຄາດ ຄະເນວ່າມີໂອກາດທີ່ຈະເກີດຂຶ້ນຕາມເຫດການສະພາບແວດລ້ອມ, ໂດຍແຕ່ລະໃບຈະຖືກເຊື່ອມດ້ວຍກິງ ກ້ານ (Branch) ເຊິ່ງເປັນຂໍ້ມູນທີ່ແຕກອອກມາຈາກໄນດຕ່າງໆ ປົງບຄືເປັນກິງກ້ານໃນການຕັດສິນໃຈວ່າ ຈະ ໃຫ້ເກີດເຫດການໄດ້ຂຶ້ນ ມີກຳເນີດຈາກໄນດເຫິງສຸດແມ່ນ ຮາກ (Root) ເຊິ່ງຈະສິ່ງມີຜົນຕໍ່ຜົນໄດ້ຮັບທີ່ ແຕກຕ່າງກັນ, ເຊັ່ນ: ຮ້ານອາຫານບ່ອນນັ່ງເຕັມ ເຮົາຈະຈອງ ຫຼືບໍ່? ຖ້າຕັດສິນໃຈວ່າບໍ່ຈອງ ເຫດການທີ່ ເກີດຂຶ້ນກໍຈະແຕກຕ່າງກັນໄປ ເຊັ່ນ: ເດີນຫາງກັບບ້ານ ຫຼື ເດີນຫາງໄປຫາຮ້ານອາຫານໃໝ່ເປັນຕົ້ນ. ຕົວຢ່າງແບບຈຳລອງ Decision Tree ສະແດງດັ່ງຮູບທີ 5.1

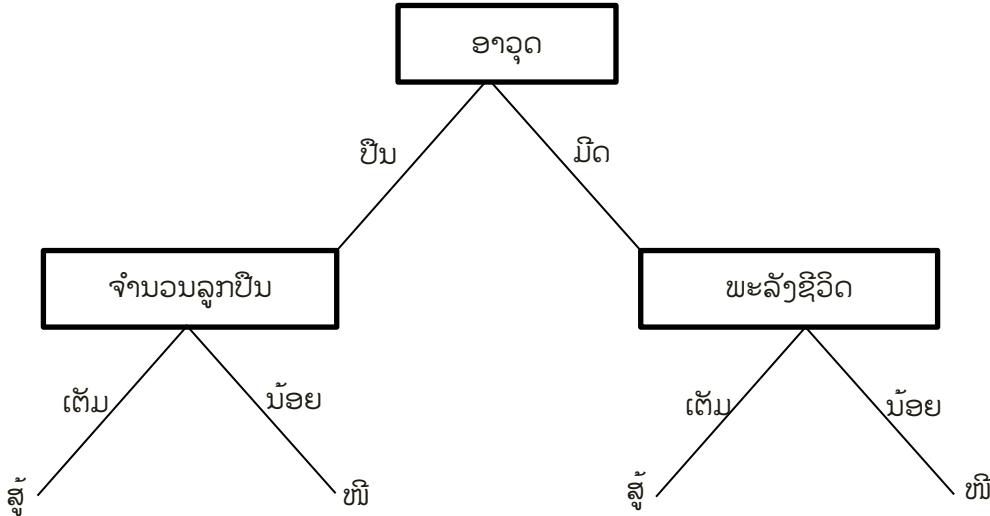


ຮູບທີ 5.1 ການສະແດງເຖິງເຫດການຮັນອາຫານເຕັມແລ້ວ

ຈາກຮູບທີ 5.1 ເປັນການສະແດງເຖິງເຫດການຮັນອາຫານເຕັມແລ້ວ ຈະຕັດສິນໃຈແນວໃດ, ເຊິ່ງເລືອກໄດ້ວ່າຈະປົງຮັນ ຫຼື ວ່າຈະຈອງໄວ້. ໃນນີ້ມີປັດໃຈເລື່ອງເວລາເຂົ້າມາກ່ຽວຂ້ອງເຮັດໃຫ້ ສາມາດຕັດສິນໃຈໄດ້ວ່າຈະຈອງໂຕະ ຫຼືບໍ່? ຈາກທີ່ກ່າວມາຈະໄຈແຍກສິ່ງທີ່ສິນໃຈໄດ້ 3 ຢ່າງຄື: ຈຳນວນລູກຄ້າ, ໄລຍະເວລາໃນການລົ່ຖ້າ ແລະ ການຈອງໂຕະ. ນອກຈາກເຫດການເຫຼົ່ານີ້ແລ້ວ ອາດມີເຫດການເພີ່ມເຕີມອື່ນທີ່ສາມາດຄາດຄະເນວ່າຈະເກີດໄດ້ອີກ ທັງນີ້ຂຶ້ນຢູ່ກັບຄວາມຕ້ອງການ ຫຼື ການຄາດຫວັງວ່າຜົນຮັບຈະອອກມາຄືແນວໃດ.

5.3. ການຮຽນຮູ້ດ້ວຍ Decision Tree

ການຮຽນຮູ້ດ້ວຍ Decision Tree ເປັນການຮຽນຮູ້ຈາກການຄາດຄະເນເຫດການຕ່າງໆທີ່ອາດເກີດຂຶ້ນ, ເຊິ່ງຈະອາໄສເງື່ອນໄຂເປັນຕົວຊ່ວຍການຕັດສິນໃຈວ່າເມື່ອເກີດເຫດການໄດ້ໜຶ່ງຂຶ້ນ ຈະແດງອອກມາແນວໃດ. ໃນການຄາດຄະເນຈະຖືກນຳສະເໜີດ້ວຍຮູບແບບການຕັດສິນໃຈທີ່ມີເງື່ອນໄຂເປັນ “ທີ່...ແລ້ວ” (if/then/else) ເຮັດໃຫ້ສາມາດຮຽນຮູ້ໄດ້ຕາມເງື່ອນໄຂທີ່ຈະເກີດຂຶ້ນໃນສະຖານະການທີ່ແຕກຕ່າງກັນເຊັ່ນ: ການຮຽນຮູ້ໃນການຫຼັບແນມທີ່ຜູ້ຫຼັບຈະມີພິດຕິກຳຄ້າຍຄືກັນເມື່ອຢູ່ໃນສະຖານະການທີ່ມີຂອບເຂດ ແລະ ຕົວປົ່ງຈຳກັດ, ໃນກໍລະນີຜູ້ຫຼັບທີ່ໃຊ້ອາວຸດປິນສາມາດຍິງໃນໄລຍະໄກໄດ້, ກ້າທີ່ຈະຕໍ່ສູ້ກັບສັດຕູເຖິງແມ່ນວ່າພະລັງຊີວິດຈະເຫຼືອໜ້ອຍກຳຕາມ, ແຕ່ຜູ້ຫຼັບທີ່ໃຊ້ອາວຸດມີດທີ່ຕ້ອງຕໍ່ສູ້ໃນໄລຍະໄກ ຈຳເປັນຕ້ອງຄໍານິ້ງເຖິງບັດໄຈດ້ານພະລັງຊີວິດຈະຕ້ອງເຫຼືອຢູ່ໃຫ້ໜ້າຍ ຫຼື ສັດຕູຕ້ອງບໍ່ເກັ່ງປານໃດ, ໃນນີ້ອາດຂຶ້ນຢູ່ກັບສະພາບແວດລ້ອມໃນຕອນນັ້ນ. ຕົວຢ່າງ Decision Tree ແບບຈ່າຍສະແດງດັ່ງຮູບ 5.2



ຮູບທີ 5.2 ສະແດງຕົວຢ່າງຂອງ Decision Tree ແບບງ່າຍ

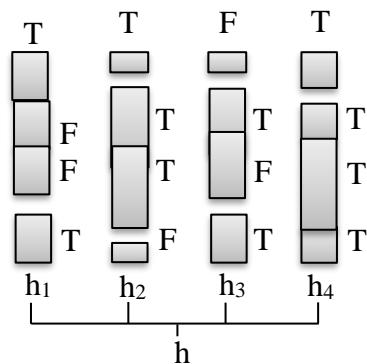
ຮູບທີ 5.2 ຜູ້ຫຼືນທີ່ໃຊ້ອາວຸດຕ່າງກັນຈະມີເງື່ອນໄຂໃນການຕັດສິນໃຈທີ່ແຕກຕ່າງກັນ ໂດຍຜູ້ຫຼືນທີ່ໃຊ້ບືນຈະຕ້ອງຄໍາໜຶ່ງເຖິງຈຳນວນລູກປິນທີ່ໃຊ້ຢູ່ໃສ່ສັດຖຸວ່າສາມາດເອົາຊະນະສັດຖຸໄດ້ ຫຼືບໍ່? ໃນຂະນະທີ່ຜູ້ຫຼືນທີ່ໃຊ້ມີດຳຈຳເປັນຕ້ອງພິຈາລະນາວ່າພະລັງຊີວິດຂອງຕົນເອງເຫຼືອພໍທີ່ຈະຕໍ່ສູ້ກັບສັດຖຸ ແລະ ເອົາຊະນະໄດ້ ຫຼືບໍ່? ຫາກທັງສອງເງື່ອນໄຂມີບໍ່ພຽງພໍຜູ້ຫຼືນຈະຕັດສິນໃຈຫຼືບໜີໝາຍກວ່າທີ່ຈະປະເສີນໜັ້າຕໍ່ສູ້.

5.4. Ensemble Learning.

ການຮຽນຮູ້ທີ່ກ່າວມາກ່ອນໜັ້ານີ້ເປັນການຮຽນຮູ້ ແລະ ການຄາດຄະເນທີ່ມີຂໍ້ມູນສະພາບແວດລ້ອມມາຈາກສົມມຸດຕິຖານດ່ວຍເຫຼົານັ້ນ ຫຼື ທີ່ເອີ້ນວ່າ “Single Hypothesis”, ແຕ່ປັນຍາປະດິດຈະຕ້ອງຮຽນຮູ້ ແລະ ວິເຄາະຂໍ້ມູນທີ່ໄດ້ມາຈາກໝາຍສົມມຸດຕິຖານເພື່ອການຕັດສິນໃຈ. ໝາຍຄວາມວ່າຂໍ້ມູນໜຶ່ງຊຸດອາດສ້າງເປັນ Decision Tree ທີ່ມີກົງກ້ານ ແລະ ໂໝດຈຳນວນໝາຍ ຈຶ່ງຈຳເປັນຕ້ອງມີການຈຳແນກ, ແຍກຍ້າຍ, ຈັດກຸ່ມລວມເຖິງຄັດເລືອກຕົວຢ່າງທີ່ດີທີ່ສຸດ ສຳລັບເອົາໄປໃຊ້ວຽກຕໍ່ໄປຕາມຂະບວນການ ຫຼື ຂັ້ນຕອນການຮຽນຮູ້. ດ້ວຍເຫດນີ້ຈຶ່ງມີທິດສະດີທີ່ຊ່ວຍສະໜັບສະໜູນໃຫ້ປັນຍາປະດິດທີ່ພັດທະນາຂຶ້ນສາມາດລວບລວມ ແລະ ເລືອກທີ່ຈະວິເຄາະສົມມຸດຕິຖານ ເພື່ອຄັດເລືອກຂໍ້ມູນທີ່ເໝາະສົມ ແລະ ມີປະສິດທິພາບໄດ້. ທິດສະດີນີ້ເອີ້ນວ່າ: “Ensemble Learning” [Russell and Norvig, 2003].

ສົມມຸດຕິຖານໃນ Ensemble Learning ຈະຖືກຈັດກຸ່ມ ແລະ ລວບລວມກ່ອນທີ່ຈະເອົາໄປໃຊ້ວິເຄາະເນື່ອງຈະຕ້ອງເຮັດການຄັດເລືອກກຸ່ມ ຫຼື ຊຸດຂອງຂໍ້ມູນຈາກສົມມຸດຕິຖານຕ່າງໆ ທີ່ມີຄວາມເປັນໄປໄດ້ ແລະ ເໝາະສົມທີ່ສຸດໃນການນຳໄປວິເຄາະ ເພື່ອຫາຂໍ້ມູນທີ່ດີທີ່ສຸດຕໍ່ໄປ. ໂດຍແຕ່ລະສົມມຸດຕິຖານຈະມີສ່ວນທີ່ສຳພັນກັນ ແລະ ສາມາດເອົາໄປໃຊ້ເປັນຂໍ້ມູນ ຫຼື ແນວທາງໃນການຮຽນຮູ້, ເຊິ່ງເປັນສ່ວນທີ່ກ່ຽວຂ້ອງກັບສິ່ງທີ່ສົນໃຈຢ່າງແທ້ຈິງ. ຫຼັກການຂອງ Ensemble Learning ນີ້ຊ່ວຍໃຫ້ການຮຽນຮູ້ມີປະສິດຕິພາບສູງຂຶ້ນ ລວມເຖິງສາມາດແກ້ໄຂບັນຫາທີ່ມີຄວາມຊັບຊັອນໝາຍໄດ້.

ສຳລັບວິທີການທີ່ນີ້ຍືມໃຊ້ທິດສະດີ Ensemble Learning ມາໃຊ້ຄື: “Boosting” ເຊິ່ງເປັນຫຼັກການທີ່ອາໄສການກຳນົດນຳໜັກໃຫ້ກັບຊຸດຂຶ້ມູນ ຫຼື ເຮັດວ່າ: “Weighted Training Set” ໂດຍໃນແຕ່ລະຊຸດຂຶ້ມູນຈະມີຄ່ານຳໜັກກຳນົດໃຫ້ຫຼາຍກ່ວ່າ ຫຼື ເທົ່າກັບ 0, ຄ່ານຳໜັກຂອງຊຸດຂຶ້ມູນໃດສູງກ່ວ່າສະແດງວ່າມີຄວາມສຳຄັນທີ່ຈະຕ້ອງພິຈາລະນາເປັນອັນດັບທຳອິດໃນຄວາມຮູ້, ອີກຢ່າງໜຶ່ງຊຸດຂຶ້ມູນດັ່ງກ່າວຍັງຢູ່ບໍ່ບອກຄວາມສຳຄັນຂອງສົມມຸດຕີຖານທີ່ກ່ຽວຂ້ອງກັບຊຸດຂຶ້ມູນນັ້ນໄດ້ອີກ. ຫຼັກການ Boosting ຈະເລີ່ມດ້ວຍການກຳນົດຄ່ານຳໜັກໃຫ້ຊຸດຂຶ້ມູນເທົ່າກັບ 1 ທັງໝົດ, ເຊິ່ງຄ່າດັ່ງກ່າວຈະເຮັດໃຫ້ເກີດຂຶ້ມູນສົມມຸດຕີຖານ h_1 ແລ້ວ ໂດຍຊຸດຂຶ້ມູນນີ້ເມື່ອເຮັດການວິເຄາະແລ້ວຈະມີທັງຂຶ້ມູນທີ່ຖືກ ແລະ ຜິດປິນກັນ ເຮັດໃຫ້ຫຼັກການວິເຄາະ h_1 ຄ່ານຳໜັກຂອງຊຸດຂຶ້ມູນຈະປູ່ງແບ່ງໄປ ມີທັງເພີ່ມຂຶ້ນ ແລະ ລຸດລົງເພື່ອໃຫ້ສາມາດຈັດກຸ່ມຊຸດໄດ້ຖືກຕ້ອງ ແລະ ເໝາະສົມ, ເຊິ່ງຈະກຳນົດການວິເຄາະຕໍ່ໄປໃນຂັ້ນຕອນຂອງ h_2 ແລະ ຂັ້ນຕອນຕໍ່ໄປຈົນກ່ວ່າຈະໄດ້ຕາມທີ່ຕ້ອງການ ດັ່ງຮູບທີ 5.3.



ຮູບທີ 5.3 ສະແດງຂັ້ນຕອນການ Boosting ທີ່ໃຊ້ທິດສະດີ Ensemble Learning

ຮູບທີ 5.3 ເປັນການດໍາເນີນການດ້ວຍຫຼັກການ Boosting ເຊິ່ງຈະເລີ່ມຈາກການກຳນົດຄ່ານຳຂັກໃຫ້ກັບຊຸດຂັ້ນມູນໃນ h_1 ເຖິງກັນທັງໝົດ ແລະ ດໍາເນີນການຕໍ່ໄປດ້ວຍການປັ້ງແປງຄ່ານຳຂັກຂອງແຕ່ລະ ຊຸດຂັ້ນມູນຈົນກ່າວຈະໄດ້ຊຸດຂັ້ນມູນທີ່ມີປະສິດຕິພາບ ແລະ ເໝາະສົມທີ່ສຸດ. ໃນການປັບຄ່ານຳຂັກສາມາດເຮັດໄດ້ຫຼາຍຮູບແບບທັງໝົດຄ່ານຳຂັກ ແລະ ເພີ່ມຄ່ານຳຂັກ. ປັດໄຈເຫຼົ້ານີ້ຊ່ວຍໃຫ້ຊຸດຂັ້ນມູນມີຄ່ານຳຂັກທີ່ເໝາະສົມທີ່ສຸດ, ທີ່ດີສະດີການລວມຊຸດຂັ້ນມູນ ຫຼື ສົມມຸດຕິຖານເຂົ້າກັນຕາມຫຼັກການຂອງ Ensemble Learning ຊ່ວຍໃຫ້ການຄາດຄະເນ ແລະ ການວິເຄາະຂັ້ນມູນມີປະສິດທິພາບສູງຂຶ້ນ, ເນື່ອງຈາກມີການປັບປຸງນຳຂັກຂອງຊຸດຂັ້ນມູນເພື່ອໃຫ້ມີຄວາມເໝາະສົມຢູ່ສະເໜີ ເຮັດໃຫ້ຂັ້ຜິດພາດ ຫຼື ຊຸດຂັ້ນມູນທີ່ສົ່ງຜົນໃຫ້ເກີດຄວາມຜິດພາດມີນ້ອຍລົງ. ດັ່ງນັ້ນ, ຈຶ່ງຊ່ວຍໃຫ້ການຝັດທະນາລະບົບການຮຽນຮູ້ສະແດງຜົນ ແລະ ສາມາດແກ້ໄຂ ພ້ອມທັງຄົນທາແວທາທີ່ເໝາະສົມກັບບັນຫາໄດ້ດີຂຶ້ນ.

5.5. ຕົວຢ່າງການປະລູກໃຊ້ Decision Tree.

ในภาคล่าง Decision Tree อาจสกัดความน่าจะเป็น ที่จำเป็นต้องคำนวณหาจากข้อมูลเดิม ที่ได้มา, เผื่องอาจได้มาจากการสัมภาษณ์ หรือ มีการกำหนดขึ้นจากแขกรับเชิญโดยกิจ. ภาระค่าความน่าจะเป็นจะพิจารณาจากข้อมูลที่ถูกใช้ในการตัดสินใจกับหัวมีดเพื่อคำนวณหาค่าของ Gain Function ต่อไป, โดยจะเอามาปูบนรากน้ำที่ตัดสินใจได้เหมาะสมสูงที่สุดเป็นฐานของ Decision

Tree ហ្មាយទីស្ថា. ផ្ទើរដែលវាឌីបែនឡើងទិន្នន័យមា, សំលប់ទិន្នន័យនឹងវេសមេដោយការបង្ហាញ Decision Tree ដើរចាប់បើជាការបែនពិនិត្យមំនា។ តាមទិន្នន័យនេះ, តាមទិន្នន័យនឹងបែនពិនិត្យការសាច់ការងារក្នុងការបង្ហាញ Decision Tree និងការបែនពិនិត្យការងារក្នុងការបង្ហាញ។

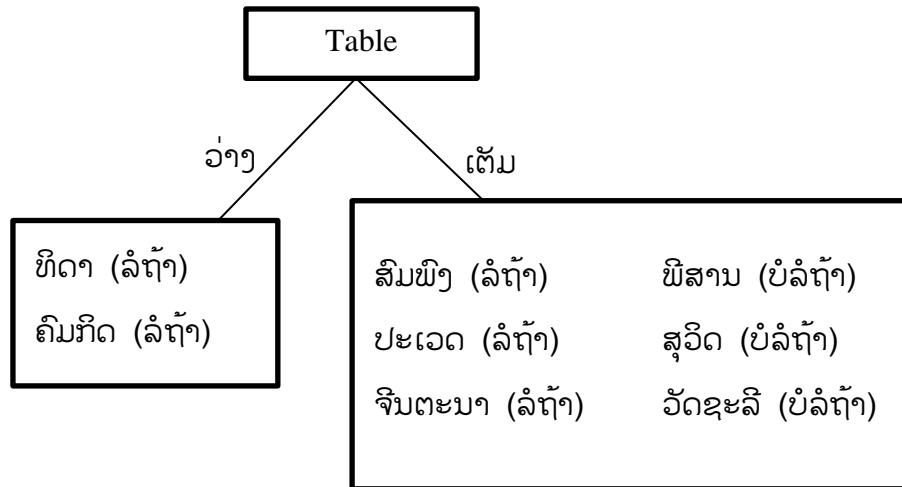
ມີການສໍາຫວັດລູກຄ້າທີ່ໃຊ້ບໍລິການຮ້ານອາຫານໃນສະຖານທີ່ແຫ່ງໜຶ່ງ ເມື່ອພົບກັບສະຖານະການທີ່ຮ້ານອາຫານທີ່ຕົນສົນໃຈເຂົ້າໃຊ້ບໍລິການບ່ອນນັ່ງເຕັມ ລູກຄ້າຈະມີພິດຕິກຳທີ່ແຕກຕ່າງກັນ ແຊ້ນ: ຫາກຮ້ານເຕັມຈະພິຈະລະນາວ່າສາມາດຈອງໄຕະໄດ້ ຫຼືບໍ່? ແລະ ໄລຍະເວລາທີ່ລົ້າດິນປານໄດ້ ຫຼືອາດຈະພິຈາລະນາຮ້ານດັ່ງກ່າວເປັນຮ້ານອາຫານປະເພດໄດ້, ລາຄາແງງຄຸມກັບອາຫານ ຫຼືບໍ່? ເປັນຕົ້ນ. ໂດຍໃຫ້ຄວາມສົນໃຈກັບພິດຕິກຳການລົ້າທີ່ຈະໃຊ້ບໍລິການໃນຮ້ານອາຫານວ່າ ປັດໄຈທີ່ແຕກຕ່າງກັນຈະສົ່ງຜົນໃຫ້ລູກຄ້າອິດຫົນລົ້າໃຊ້ບໍລິການ ຫຼືບໍ່?. ສໍາລັບຂໍ້ມູນຈາກລູກຄ້າທັງໝົດ 8 ຄົນ ແຊ້ງມີຂໍ້ມູນດີບສະແດງໃນຕາຕະລາງທີ່ 5.2

ລູກຄ້າ	ສະຖານະຂອງໂຕະ	ລາຄາ	ຮັບອາຫານ	ໄລຍະເວລາລຳຖາ(ນາທີ)	ລູກຄ້າລຳຖາ
ສິມພິງ	ເຕັມ	ແພງ	ຍື່ປຸ່ນ	10-30	ລຳຖາ
ຈິນຕະນາ	ເຕັມ	ຖືກ	ໄທ	10-30	ລຳຖາ
ຟິສານ	ເຕັມ	ແພງ	ໄທ	10-30	ບໍລິຖາ
ສຸວິດ	ເຕັມ	ແພງ	ຍື່ປຸ່ນ	ໝາຍກວ່າ 30	ບໍລິຖາ
ປະເວດ	ເຕັມ	ແພງ	ອິຕາລີ	0-10	ລຳຖາ
ຫົດາ	ວ່າງ	ຖືກ	ໄທ	10-30	ລຳຖາ
ວັດຊະລີ	ເຕັມ	ແພງ	ອິຕາລີ	10-30	ບໍລິຖາ
ຄືມກິດ	ວ່າງ	ແພງ	ໄທ	0-10	ລຳຖາ

ຕາຕະລາງທີ 5.2 ສະແດງຂໍ້ມູນດີບຂອງການສໍາຫວັດຄວາມອິດທຶນຂອງລົກຄ້າທີ່ລໍຖ້າໃຊ້ບໍລິການ.

ຈາກຕາຕະລາງ 5.2 ເຮັດໃຫ້ເຄີຍຮູ່ວ່າລູກຄ້າແຕ່ລະຄົມມີຄວາມຕ້ອງການ ແລະ ພິດຕິກຳໃນສະຖານະການໜຶ່ງໆ ຫີ້ແຕກຕ່າງກັນ, ໂດຍຈະໃຫ້ຄວາມສົນໃຈກັບພິດຕິກຳການລົ້າຂອງລູກຄ້າວ່າຈະສາມາດລົ້າເຂົ້າໃຊ້ບໍລິການຮ້ານຄ້ານັ້ນໄດ້ ຫຼືບໍ່? ໃນປັດໄຈທີ່ແຕກຕ່າງກັນ, ເຊິ່ງປັດໄຈດັ່ງກ່າວເປັນຊຸດຂໍ້ມູນທີ່ຈຳເປັນຕົງພິຈາລະນາໃນການຄໍານວນຫາຄ່າ Gain Function ໄດ້ແກ່ສະຖານະຂອງໂຕະ (Table), ລາຄາອາຫານ (Price), ປະເພດຮ້ານອາຫານ (Type) ແລະ ໄລຍະເວລາໃນການລົ້າ (Time) ສະແດງເປັນ Decision Tree ໄດ້ດັ່ງນີ້.

ສະຖານະຂອງໄຕະ (Table)



ຮູບທີ 5.4 ແບບຈຳລອງສະຖານະຂອງໄຕະ

ຈາກແບບຈຳລອງສະຖານະຂອງໄຕະ ເມື່ອແກນໃນ Gain Function ຈະໄດ້ດັ່ງນີ້:

$$E(s) = -\frac{5}{8} \log_2 \frac{5}{8} - \frac{3}{8} \log_2 \frac{3}{8}$$

ດັ່ງນັ້ນຈະໄດ້

$$\text{Gian(price)} = \left(-\frac{5}{8} \log_2 \frac{5}{8} - \frac{3}{8} \log_2 \frac{3}{8} \right) - \left(\frac{2}{8} \left(-\frac{2}{2} \log_2 \frac{3}{2} \right) + \frac{6}{8} \left(-\frac{3}{6} \log_2 \frac{3}{6} - \frac{3}{6} \log_2 \frac{3}{6} \right) \right)$$

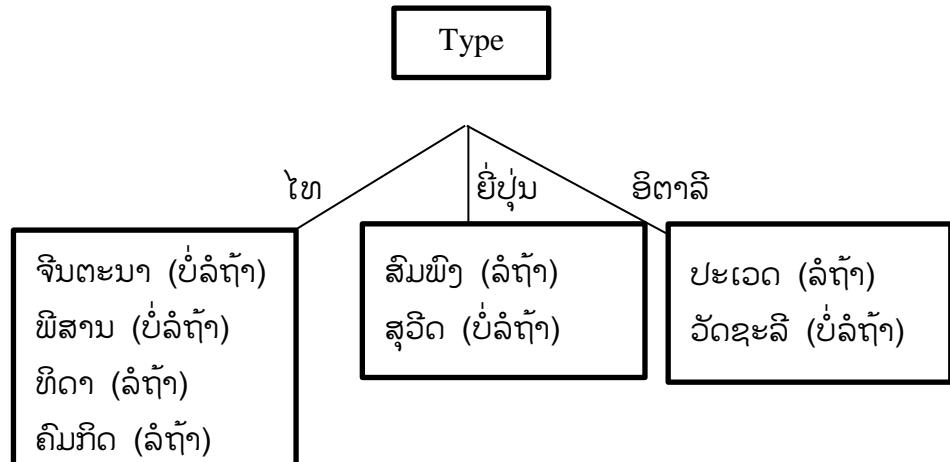
$$\text{Gian(price)} = \left(-\frac{5}{8} \log_2 \frac{5}{8} - \frac{3}{8} \log_2 \frac{3}{8} \right) - \left(\frac{1}{4} (-\log_2 1) + \frac{3}{4} \left(-\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2} \right) \right)$$

$$\text{Gian(price)} = (0.4237 + 0.5306) - \left(-\frac{1}{4}(0) + \frac{3}{4}(0.5 + 0.5) \right)$$

$$\text{Gian(price)} = (0.9543) - \left(\frac{3}{4}(1) \right)$$

$$\text{Gian(price)} = 0.9543 - 0.75 = 0.2043$$

ປະເພດຮ້ານອາຫານ (Type)



ຮູບທີ 5.5 ແບບຈຳລອງປະເພດຮ້ານອາຫານ

ຈາກແບບຈຳລອງປະເພດຮ້ານອາຫານ ເມື່ອແກນໃນ Gain Function ຈະໄດ້ດັ່ງນີ້:

$$E(s) = -\frac{5}{8} \log_2 \frac{5}{8} - \frac{3}{8} \log_2 \frac{3}{8}$$

ດັ່ງນັ້ນຈະໄດ້

$$\begin{aligned} \text{Gain}(Type) &= \left(-\frac{5}{8} \log_2 \frac{5}{8} - \frac{3}{8} \log_2 \frac{3}{8} \right) \\ &\quad - \left(\frac{4}{8} \left(-\frac{3}{4} \log_2 \frac{3}{4} - \frac{1}{4} \log_2 \frac{1}{4} \right) + \frac{2}{8} \left(-\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2} \right) \right. \\ &\quad \left. + \frac{2}{8} \left(-\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2} \right) \right) \end{aligned}$$

$$\begin{aligned} \text{Gain}(Type) &= \left(-\frac{5}{8} \log_2 \frac{5}{8} - \frac{3}{8} \log_2 \frac{3}{8} \right) \\ &\quad - \left(\frac{1}{2} \left(-\frac{3}{4} \log_2 \frac{3}{4} - \frac{1}{4} \log_2 \frac{1}{4} \right) + \frac{1}{4} \left(-\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2} \right) \right. \\ &\quad \left. + \frac{1}{4} \left(-\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2} \right) \right) \end{aligned}$$

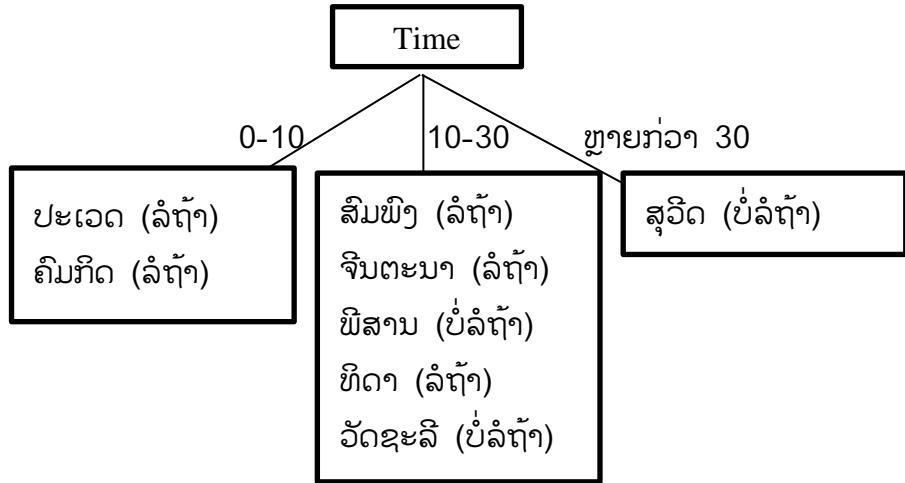
$$\text{Gain}(Type) = (0.4237 + 0.5306) - \left(\frac{1}{2} (0.3112 + 0.5) + \frac{1}{4} (0.5 + 0.5) \frac{1}{4} (0.5 + 0.5) \right)$$

$$\text{Gain}(Type) = (0.9543) - \left(\frac{1}{2} (0.8112) + \frac{1}{4} (1) \frac{1}{4} (1) \right)$$

$$\text{Gain}(Type) = (0.9543) - (0.4056 + 0.25 + 0.25)$$

$$\text{Gain}(Type) = 0.9543 - 0.9056 = 0.0487$$

ໄລຍະເວລາໃນການລົ້າ (Time).



ຮູບທີ 5.6 ແບບຈຳລອງໄລຍະເວລາໃນການລົ້າ

ຈາກແບບຈຳລອງໄລຍະເວລາໃນການລົ້າ ເມື່ອແກນໃນ Gain Function ຈະໄດ້ດັ່ງນີ້:

$$E(s) = -\frac{5}{8} \log_2 \frac{5}{8} - \frac{3}{8} \log_2 \frac{3}{8}$$

ດັ່ງນັ້ນຈະໄດ້

$$\begin{aligned} \text{Gain}(Time) &= \left(-\frac{5}{8} \log_2 \frac{5}{8} - \frac{3}{8} \log_2 \frac{3}{8} \right) \\ &\quad - \left(\frac{2}{8} \left(-\frac{2}{2} \log_2 \frac{2}{2} \right) + \frac{5}{8} \left(-\frac{3}{5} \log_2 \frac{3}{5} - \frac{2}{5} \log_2 \frac{2}{5} \right) + \frac{1}{8} \left(-\frac{1}{1} \log_2 \frac{1}{1} \right) \right) \end{aligned}$$

$$\begin{aligned} \text{Gain}(Time) &= \left(-\frac{5}{8} \log_2 \frac{5}{8} - \frac{3}{8} \log_2 \frac{3}{8} \right) \\ &\quad - \left(\frac{1}{4} (-\log_2 1) + \frac{5}{8} \left(-\frac{3}{5} \log_2 \frac{3}{5} - \frac{2}{5} \log_2 \frac{2}{5} \right) + \frac{1}{8} (-\log_2 1) \right) \end{aligned}$$

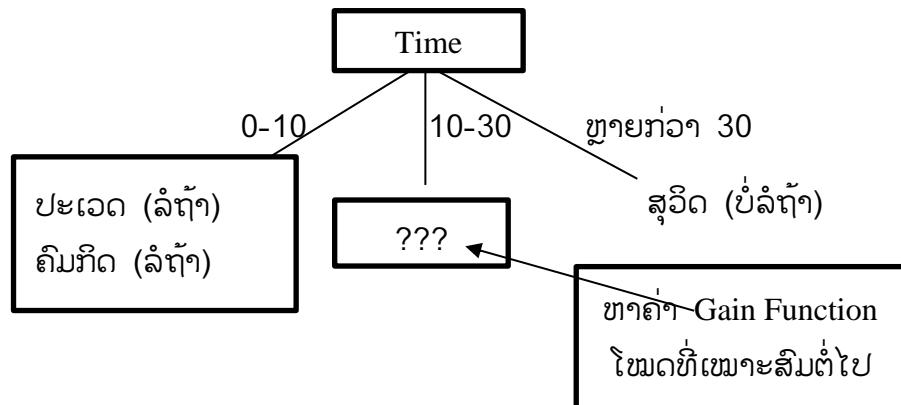
$$\text{Gain}(Time) = (0.4237 + 0.5306) - \left(\frac{1}{4}(0) + \frac{5}{8}(0.4421 + 0.5287) + \frac{1}{8}(0) \right)$$

$$\text{Gain}(Time) = (0.9543) - \left(\frac{5}{8}(0.9708) \right)$$

$$\text{Gain}(Time) = 0.9543 - 0.6067 = 0.39708$$

ຈາກການຄຳນວນຄ່າ Gain Function ຂອງແຕ່ລະຊຸດຂໍ້ມູນຈະໄດ້ຄ່າທີ່ແຕກຕ່າງກັນໄປ, ໂດຍຄ່າທີ່ຫຼາຍທີ່ສຸດຈະສາມາດບົງບອກໄດ້ວ່າຊຸດຂໍ້ມູນນັ້ນມີຄວາມເໝາະສົມໃນການນຳມາເປັນໂໜດເລີ່ມຕົ້ນ ໃນການສ້າງ Decision Tree ເຊິ່ງກໍ່ແມ່ນ Gain (Time) ທີ່ມີຄ່າຫຼາຍທີ່ສຸດ, ຈາກນັ້ນຈຶ່ງພິຈາລະນາໂໜດທີ່ເໝາະສົມໃນອັນດັບຕໍ່ໄປໂດຍການຄຳນວນຫາ Gain Function ເຊິ່ນດູວກັນກັບຂັ້ນຕອນທີ່ຜ່ານມາ, ແຕ່ຈະພິຈາລະນາພຽງໂໜດທີ່ເຫຼືອ ເຊິ່ງຈະເຊື່ອມຕໍ່ກັບໂໜດທີ່ເລີ່ມຕົ້ນ ໂດຍພິຈາລະນາຈາກກົງກັນທີ່ຍັງມີ

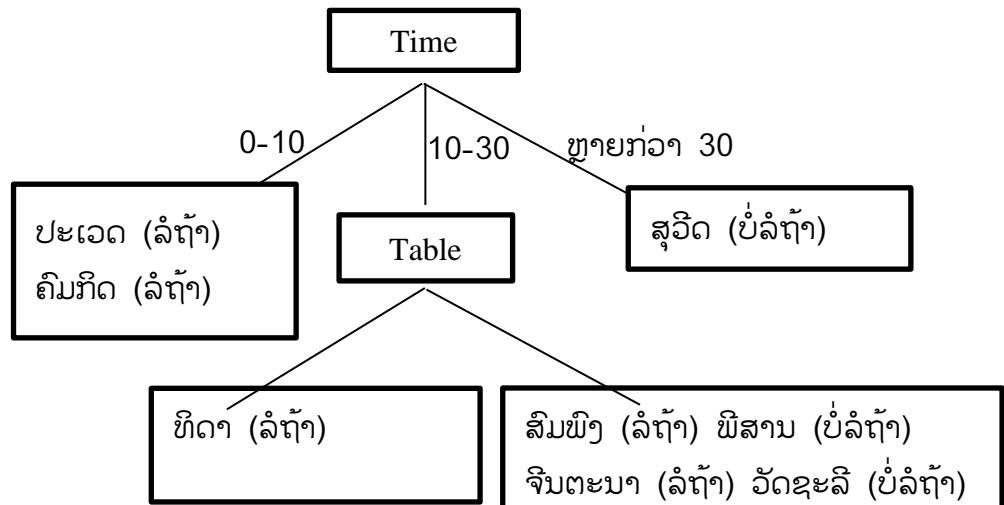
ຄວາມຊັບຊັນຂອງຂໍ້ມູນ ຫຼື ຍັງຈໍາແນກໄດ້ບໍ່ໝົດ, ໂໝາດນັ້ນຍັງຕ້ອງແຕກກີ່ງກຳນົມຕໍ່ໄປເພື່ອແນກຂໍ້ມູນທີ່ແຕກຕ່າງກຳນອກເປັນກຸ່ມທີ່ເໝາະສິມ. (ດັ່ງຮູບທີ 5.7 ໂໝາດເຄື່ອງກາງຍັງມີຄວາມຕ້ອງການຂອງລູກຄ້າ ລົມຖາ ແລະ ບໍ່ລົມຖາ ປຶນກັນຢູ່)



ຮູບທີ 5.7 ສະແດງການຫາຄ່າໂໝາດລົມຖາຕໍ່ໄປໃນ Decision Tree

ດັ່ງນັ້ນຈຶ່ງຕ້ອງພິຈາລະນາຄ່າ Gain Function ຂອງຊຸດຂໍ້ມູນທີ່ເຫຼືອ ດັ່ງນີ້:

ສະຖານະຂອງໂຕະ (Table)



ຮູບທີ 5.8 ແບບຈຳລອງປະລິມານໂຕະວ່າງ

ຈາກແບບຈຳລອງປະລິມານໂຕະວ່າງ ເມື່ອແກນໃນ Gain Function ຈະໄດ້ດັ່ງນີ້:

$$E(s) = -\frac{3}{5} \log_2 \frac{3}{5} - \frac{2}{5} \log_2 \frac{2}{5}$$

ດັ່ງນັ້ນຈະໄດ້

$$\text{Gain(Table)} = \left(-\frac{3}{5} \log_2 \frac{3}{5} - \frac{2}{5} \log_2 \frac{2}{5} \right) - \left(\frac{1}{5} \left(-\frac{1}{1} \log_2 \frac{1}{1} \right) + \frac{4}{5} \left(-\frac{2}{4} \log_2 \frac{2}{4} - \frac{2}{4} \log_2 \frac{2}{4} \right) \right)$$

$$\text{Gain(Table)} = \left(-\frac{3}{5} \log_2 \frac{3}{5} - \frac{2}{5} \log_2 \frac{2}{5} \right) - \left(\frac{1}{5} (-\log_2 1) + \frac{4}{5} \left(-\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2} \right) \right)$$

$$\text{Gain(Table)} = (0.4421 + 0.5257) - \left(\frac{1}{5}(0) + \frac{4}{5}(0.5 + 0.5) \right)$$

$$\text{Gain(Table)} = (0.9708) - \left(\frac{4}{5}(1) \right)$$

$$\text{Gain(Time)} = 0.9543 - 0.80 = 0.1708$$

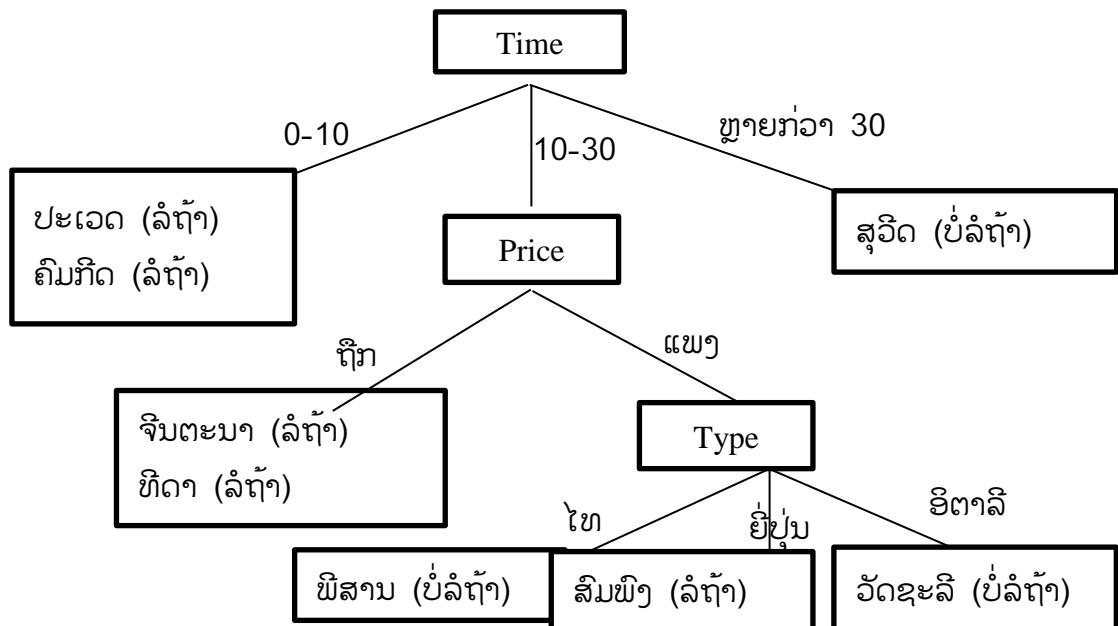
ນອກຈາກຊຸດຂໍ້ມູນປະລິມານໂຕະວ່າງແລວຈໍາເປັນຕ້ອງຄໍານວນຄ່າ Gain Function ຂອງຊຸດຂໍ້ມູນອື່ນດ້ວຍ (Price, Type) ເຊິ່ງໄດ້ຄ່າດັ່ງນີ້:

$$\text{Gain (Table)} = 0.1708$$

$$\text{Gain (Price)} = 0.4199$$

$$\text{Gain (Type)} = 0.4199$$

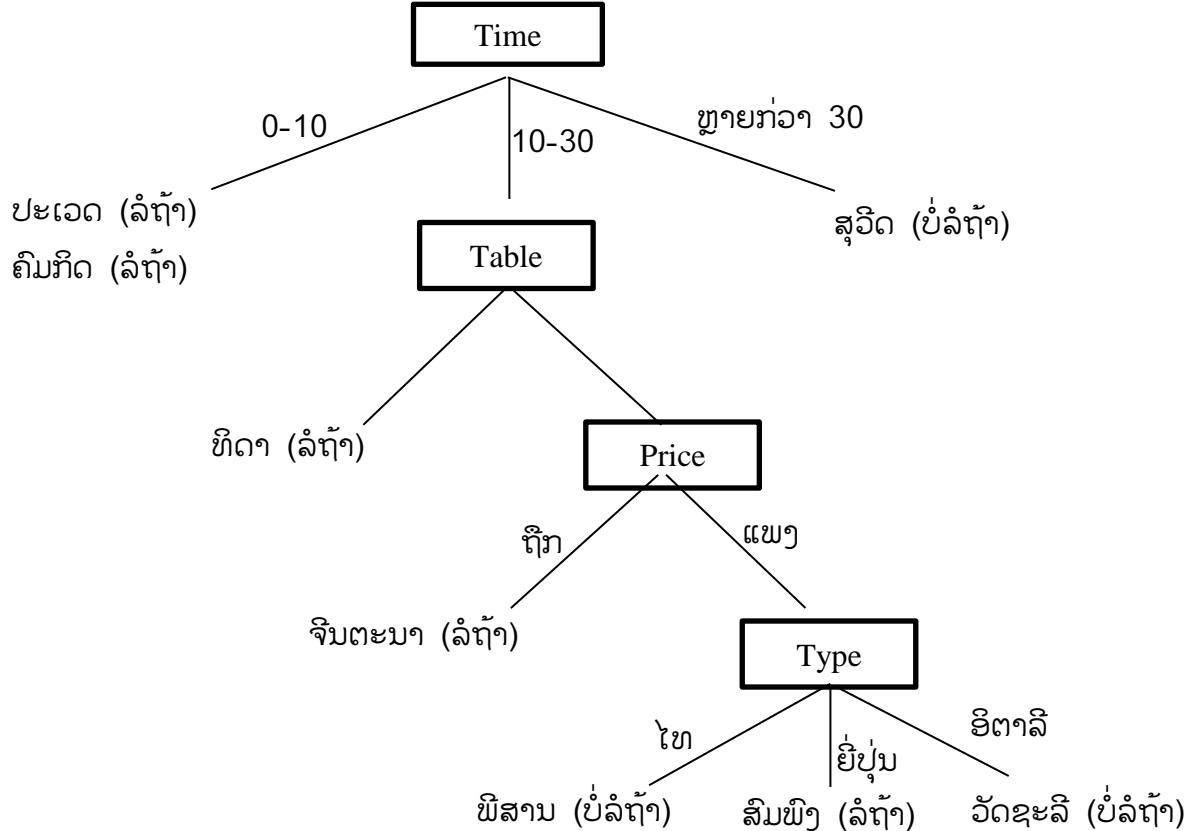
ໃນກໍລະນີທີ່ຄ່າຂອງ Gain Function ທີ່ເພົ່າສົມເຫຼົາກັນຈະສາມາດເລືອກຊຸດຂໍ້ມູນຊຸດໄດ້ກໍໄດ້ເນື້ອງຈາກຊຸດຂໍ້ມູນທັງສອງສາມາດຄໍານວນຂໍ້ມູນໄດ້ຢ່າງສົມບູນຄືກັນ ພຽງແຕ່ຈະມີໂຄງສ້າງຂອງ Decision Tree ທີ່ແຕກຕ່າງກັນເຫຼົານັ້ນສໍາລັບຕົວຢ່າງນີ້ຈະເລືອກ Price ເປັນໄໝດຕໍ່ໄປ ເຊິ່ງຈະໄດ້ Decision Tree ດັ່ງຮູບທີ່ 5.9



ຮູບທີ່ 5.9 ສະແດງ Decision Tree ທີ່ສົມບູນຂອງຕົວຢ່າງທີ່ 5.3

ຮູບທີ່ 5.9 ຈະເປັນ Decision Tree ທີ່ມີຄວາມສົມບູນແລວ ເນື້ອງຈາກສາມາດໄຈແຍກຂໍ້ມູນທີ່ສົນໃຈອອກຈາກຊຸດຂໍ້ມູນທັງໝົດໄດ້ຢ່າງຈະແຈ້ງ, ເຖິງວ່າຈະມີໄໝດບໍ່ຄົບຕາມຈໍານວນຂອງຊຸດຂໍ້ມູນທັງໝົດທີ່ພິຈາລະນາກໍ່ຕາມ. ເນື້ອງຈາກ Decision Tree ທີ່ໄດ້ບໍ່ຈໍາເປັນຕ້ອງເອົາໄໝດ Table ມາພິຈາລະນາ. ດ້ວຍເຫດນີ້ຈຶ່ງມີຄວາມຊັບຊັອນນ້ອຍກ່ວາ Decision Tree ທີ່ເອົາໄໝດ Table ມາຮ່ວມ

ພິຈາລະນາ. ໃນຄວາມເປັນຈິງອາດມີ Decision Tree ທີ່ສອດຄ່ອງກັບຊຸດຂໍ້ມູນທັງໝົດໄດ້ຫຼາຍກ່ວາໜຶ່ງຮູບແບບ, ເຊິ່ງອາດເປັນ Decision Tree ທີ່ງ່າຍຕໍ່ການພິຈາລະນາຂໍ້ມູນ ຫຼື ອາດຊັບຊັອນກ່ວາ Decision Tree ເກົ່າກຳໄດ້ ດັ່ງຮູບທີ 5.10

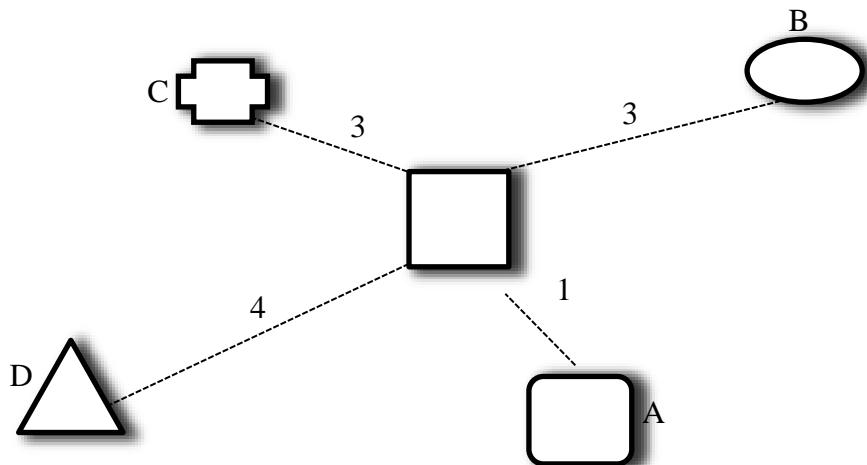


ຮູບທີ 5.10 ສະແດງຕົວຢ່າງ Decision Tree ອີກໍ່ງຮູບແບບທີ່ສອດຄ່ອງກັບ
ຊຸດຂໍ້ມູນຂອງຕົວຢ່າງທີ 5.3

ຮູບທີ 5.9 ແລະ 5.10 ຈະເຫັນໄດ້ວ່າ Decision Tree ທັງໝົດແຕກຕ່າງກັນ, ໃນຮູບທີ 5.10 ຈະມີຈຳນວນໂພມຂອງໃບຫຼາຍກ່ວາ ແລະ ສາມາດຈຳແນກຂໍ້ມູນອອກຈາກຊຸດຂໍ້ມູນໄດ້ຄືກັນກັບຮູບທີ 5.9 ແຕ່ຄວາມຊັບຊັອນ ຫຼື ຈຳນວນໃບຫຼາຍກ່ວາ ອາດເຮັດໃຫ້ການຕັດສິນໃຈຊ້າລົງໄດ້. ຖ້າພິຈາລະນາຮູບທີ 5.9 ແລ້ວຈະເຫັນໄດ້ວ່າປັດໄຈສຳຄັນທີ່ສຸດຂອງພິດຕິກຳການລົ້ຖ້າຂອງລູກຄ້າທັງ 8 ຄົນ ຈະເປັນເລື່ອງຂອງເວລາ, ດັ່ງນັ້ນສະຖານະຂອງໂຕະຈຶ່ງບໍ່ຈຳເປັນຖ້າລູກຄ້າຮູ້ໄລຍະເວລາທີ່ແນ່ນອນວ່າຕ້ອງລົ້ຖ້າດີນປານໄດ. ເມື່ອພິຈາລະນາທີ່ໂພມທຳອິດລົງມາຄືລາຄາ ຖໍ່ວ່າເປັນປັດໄຈທີ່ຊ່ວຍຈຳແນກຂໍ້ມູນໄດ້ມີປະສິດທິພາບຫຼາຍກ່ວາຊຸດຂໍ້ມູນສະຖານະຂອງໂຕະ, ເມື່ອງຈາກວ່າສາມາດຈຳແນກລູກຄ້າທີ່ມີຄວາມຕ້ອງການລົ້ຖ້າໄດ້ 2 ຄົນ ໃນຄະນະທີ່ໂພມສະຖານະຂອງໂຕະສາມາດຈຳແນກໄດ້ພູງ 1 ຄົນເທົ່ານັ້ນ. ຖ້າເປັນລະບົບທີ່ມີຂໍ້ມູນຈຳນວນຫຼາຍອາດຈະເຮັດໃຫ້ເສຍເວລາໄປໂດຍບໍ່ມີປະໂຫຍດ. ດັ່ງນັ້ນໃນການສ້າງ Decision Tree ຈຳເປັນຕ້ອງຄໍານວນຫາຊຸດຂໍ້ມູນທີ່ເພົາະສົມທີ່ສຸດ ເພື່ອໃຫ້ໄດ້ Decision Tree ທີ່ສົມຂູນ ແລະ ມີປະສິນຕິພາບເມື່ອເອົາໄປໃຊ້ໃນລະບົບການຮຽນຮູ້ຂອງປັນຍາປະດິດ.

5.6. Nearest Neighbor Classification

ເປັນການຮຽນຮູ້ປະເພດ Unsupervised Learning ເຊິ່ງເປັນການຈຳແນກ ຫຼື ຈັດກຸມທີ່ມີວິທີການບໍ່ຂັບຂັນ, ໂດຍພິຈາລະນາຊຸດຂໍ້ມູນທີ່ໄກຄົງກ່ວາຄື: ຂໍ້ມູນມີຄ່າທີ່ໄກຄົງກັບຄ່າຂອງຂໍ້ມູນທີ່ພິຈາລະນາໝາຍທີ່ສຸດ, ໃນນີ້ຄ່າຄວາມໄກຄົງຈະໝາຍເຖິງໄລຍະຫາງ (Distance) ທີ່ມີຄ່ານັ້ນອຍທີ່ສຸດລະຫວ່າງຊຸດຂໍ້ມູນກັບຂໍ້ມູນທີ່ພິຈາລະນາ ຂໍ້ມູນດັ່ງກ່າວນີ້ເອີ້ນວ່າ: “Nearest Neighbor” ດັ່ງຮູບທີ 5.11



ຮູບທີ 5.11 ສະແດງຕົວຢ່າງໄລຍະເວລາລະຫວ່າງຊຸດຂໍ້ມູນ [Jones, 2008]

ຮູບທີ 5.11 ຊຸດຂໍ້ມູນທີ່ມີໄລຍະຫາງໄກກັບສູນກາງໝາຍທີ່ສຸດຄື A ເຊິ່ງມີໄລຍະເປັນ 1 ໃນທີ່ດີຂອງ Nearest Neighbor ການຈຳແນກຊຸດຂໍ້ມູນທີ່ມີໄລຍະຫາງ 1 ຈະເອີ້ນວ່າ: “1NN (One Nearest Neighbor)” ໂດຍໄລຍະຫາງຂອງຂໍ້ມູນນັ້ນສາມາດກຳນົດໄດ້ວ່າຕ້ອງການໜ້ອຍໝາຍເຖິງໄດ, ດ້ວຍເຫດນີ້ຈຶ່ງມີການຈຳແນກທີ່ເອີ້ນວ່າ: “k-NN” ເຊິ່ງ k ແທນດ້ວຍຄ່າໄລຍະຫາງລະຫວ່າງຂໍ້ມູນທີ່ຕ້ອງການ ສໍາລັບການຫາຄ່າໄລຍະຫາງຈະໃຊ້ສົມຜົນຈາກທິດສະດີການວັດຄ່າຂອງ Euclidean ດັ່ງນີ້.

$$d = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}$$

ສົມຜົນດັ່ງກ່າວເປັນການຫາໄລຍະຫາງລະຫວ່າງຊຸດຂໍ້ມູນ ໂດຍແທນຄ່າທີ່ບັນຈຸຢູ່ພາຍໃນຊຸດຂໍ້ມູນທັງສອງ, ເຊິ່ງ p ແມ່ນຄ່າໃນຊຸດຂໍ້ມູນທີ່ຕ້ອງການຈຳແນກ ມີຄ່າຕັ້ງແຕ່ $(p_1, p_2, p_3, \dots, p_n)$ ແລະ q ແມ່ນຄ່າໃນຊຸດຂໍ້ມູນຂ້າງຄົງທີ່ນຳມາພິຈາລະນາ ມີຄ່າຕັ້ງແຕ່ $(q_1, q_2, q_3, \dots, q_n)$ ຂຶ້ນຢູ່ກັບຈຳນວນຂໍ້ມູນທີ່ຢູ່ພາຍໃນຊຸດຂໍ້ມູນນັ້ນ ດັ່ງຕົວຢ່າງທີ 5.4

ຕົວຢ່າງທີ 5.4

ກຳນົດໃຫ້ມີການຈຳແນກສັດ 5 ຂະນິດ ຈຳແນກຄຸນສົມບັດ ທີ່ສາມາດບິ່ງບອກເຖິງສາຍພັນໄດ້ ດັ່ງນີ້:

1				1	1		ສັດເຕິ່ງປົກເຕິ່ງນຳ
1		1	1		1		ສັດຈຳພວກນິກາ
1	1	1					ສັດລົງລູກດ້ວຍນຳນິມ
1			1		1		ສັດເລືອຍຄານ
1			1		1	1	ສັດຈຳພວກປາ

ມີຊັດຂໍ້ມູນຂອງສັດຊະນິດທີ່ກັບພົງຂໍ້ມູນດ້ານຄຸນສົມບັດ ດັ່ງນີ້

โดยกำหนดว่าสัด比นิดนี้เป็นสัดที่ไว้คู่กับสายพันได้หลายเท่าๆ 1NN จำกัดด้วยมุมของสัดปิดเศษๆ จะเอามาใช้หาค่าໄลยละเอียดของดูดขั้มูลสัดห้า 5 อะนิด ด้วยสิ่ง Euclidean เมื่อแทนค่าในแต่ละดูดขั้มูลได้ดังนี้:

ໄລຍະທາງຂອງຊຸດຂໍ້ມູນກີບ (d_1)

$$d_1 = \sqrt{(0)^2 + (1-1)^2 + (0)^2 + (0)^2 + (0)^2 + (0)^2 + (0-1)^2 + (1-1)^2 + (0)^2} = \sqrt{1} = 1$$

ໄລຍະທາງຂອງຊຸດຂໍ້ມູນເປັດ (d_2)

$$d_2 = \sqrt{(0)^2 + (1-1)^2 + (0)^2 + (0-1)^2 + (0)^2 + (0-1)^2 + (0)^2 + (1-1)^2 + (0)^2} \\ = \sqrt{1+1} = \sqrt{2} = 1.414$$

ໄລຍະທາງຂອງຊຸດຂໍ້ມູນເຈຍ (d₃)

$$d_3 = \sqrt{(0-1)^2 + (1-0)^2 + (0-1)^2 + (0-1)^2 + (0)^2 + (0)^2 + (0)^2 + (1-0)^2 + (0)^2} \\ = \sqrt{1+1+1+1+1} = \sqrt{5} = 2.236$$

ໄລຍະທາງຂອງຊຸດຂໍ້ມູນງູ (d₄)

$$d_4 = \sqrt{(0)^2 + (1-1)^2 + (0)^2 + (0)^2 + (0-1)^2 + (0)^2 + (0-1)^2 + (1-0)^2 + (0)^2} \\ = \sqrt{1+1+1} = \sqrt{3} = 1.732$$

ໄລຍະທາງຂອງຊຸດຂໍ້ມູນປາຊາມອນ (d₅)

$$d_5 = \sqrt{(0)^2 + (1-1)^2 + (0)^2 + (0)^2 + (0-1)^2 + (0)^2 + (0-1)^2 + (1-0)^2 + (0-1)^2} \\ = \sqrt{1+1+1+1} = \sqrt{4} = 2$$

ຈາກຄ່າຂອງໄລຍະທາງທີ່ໄດ້ຈາກຊຸດຂໍ້ມູນສັດທ່າງ 5 ຊະນິດ ໄລຍະທາງທີ່ມີຄ່າໜ້ອຍທີ່ສຸດຄື d₁ ສະແດງວ່າສັດປິດສະໜາມີຂໍ້ມູນທີ່ໄກ້ຄົງງັບຊຸດຂໍ້ມູນຂອງກີບ. ຈຶ່ງສະຫຼຸບໄດ້ວ່າ: ສັດປິດສະໜາຊະນິດນີ້ ມີສາຍພັນເປັນ ສັດເຕັ້ງບິກເຕັ້ງນັ້ນ.

ຈາກຕົວຢ່າງທີ 5.4 ເປັນການຄຳນວນຫາໄລຍະທີ່ໄກ້ທີ່ສຸດຄື ຈະມີຊຸດຂໍ້ມູນຂ້າງຄ່າງທີ່ມີໄລຍະທ່າງນ້ອຍທີ່ສຸດພຽງໜຶ່ງຊຸດຂໍ້ມູນເທົ່ານັ້ນ, ໃນກໍລະນີທີ່ຕ້ອງການຊຸດຂໍ້ມູນທີ່ມີຄ່າໄກ້ຄົງຫຼາຍກ່ວາໜຶ່ງຊຸດຂໍ້ມູນ ຫຼື k ຊຸດ, ການຈຳແນກຫາຊຸດຂໍ້ມູນປະເພດນີ້ຈະເຮື່ອງວ່າ: “k-NN” ດັ່ງຕໍ່ໄປນີ້.

ຕົວຢ່າງທີ 5.5

ກຳນົດໃຫ້ມີການຈຳແນກສັດ 5 ຊະນິດ ຈຳແນກຄຸນສົມບັດ ທີ່ສາມາດບິ່ງບອກເຖິງສາຍພັນໄດ້ດັ່ງນີ້:

	ສາຍພັນ							
ກີບ	1				1	1		ສັດເຕັ້ງບິກເຕັ້ງນັ້ນ
ເປັດ	1		1		1			ສັດຈຳພວກນິກ
ເຈຍ	1	1	1					ສັດລົງງລູກດ້ວຍນັ້ນນິມ
ງູ	1			1		1		ສັດເລືອຍຄານ
ປາຊາມອນ	1			1		1		ສັດຈຳພວກປາ
ແຂ້	1			1		1		ສັດເລືອຍຄານ
ແມວ	1	1	1		1			ສັດລົງງລູກດ້ວຍນັ້ນນິມ
ນິກກະຈອກເທດ	1			1				ສັດຈຳພວກນິກ

ມີຊຸດຂໍ້ມູນຂອງສັດຊະນິດທີ່ງຮູ້ພຽງຂໍ້ມູນດ້ານຄຸນສົມບັດ ດັ່ງນີ້:

ສັດປິດສະໜາ		1	1		1				???	ສາຍພັນ
ອອກຈາກປັບປຸງ	ຈອກແຈ້ງປັບປຸງ	ລົງຈາກດ້ວຍໜົນ	ປະເມີນການແກ່ໄວ	ຜູ້ໃຫ້ຜົນ	ກົດຕົວວັດ	ກົດຕົວ	ກົດຕົວ	ກົດຕົວ	ກົດຕົວ	ທາງໃຈດ້ວຍທຽບ

ຕ້ອງການຮູ້ວ່າສັດຊະນິດນີ້ເປັນສັດທີ່ໄກຄູ່ງກັບສາຍພັນໄດ້ແດ່ຢ່າງນອຍ 3 ຊະນິດ (3NN) ທາຄາໄລຍະຫາງຂອງແຕ່ລະຊຸດຂໍ້ມູນ (ຄືຕົວຢ່າງທີ 5.4) ຈະໄດ້ຜົນຮັບດັ່ງນີ້:

$$\text{ກົບ } (d_1) = 2.449$$

$$\text{ປາຊາມອນ } (d_5) = 2.645$$

$$\text{ເປັດ } (d_2) = 1.732$$

$$\text{ແຂ້ } (d_6) = 2.236$$

$$\text{ເຈຍ } (d_3) = 1.414$$

$$\text{ແມວ } (d_7) = 1$$

$$\text{ງູ } (d_4) = 2.236$$

$$\text{ນິກາກະຈອກເທັດ } (d_8) = 1.414$$

ຈາກທ່າໄລຍະຫາງທີ່ໄດ້ຂອງທັງ 8 ຊຸດຂໍ້ມູນ ຈະເລືອກທ່າໄລຍະຫາງທີ່ນ້ອຍທີ່ສຸດ 3 ອັນດັບໄດ້ແກ່ ແມວ (d₇) = 1, ເຈຍ (d₃) = 1.414 ແລະ ນິກາກະຈອກເທັດ (d₈) = 1.414 ດັ່ງນັ້ນສັດປິດສະໜາດັ່ງກ່າວມີຄວາມໄກຄູ່ງກັບສາຍພັນສັດລົງລູກດ້ວຍນຳນິມ ແລະ ສັດຈຳພວກນິກ. ແຕ່ວ່າທັງ 3 ຊຸດຂໍ້ມູນທີ່ໄດ້ມາມີ 2 ຊຸດຂໍ້ມູນທີ່ມີສາຍພັນຄືກັນ, ຈຶ່ງສາມາດສະຫຼຸບຈາກຂໍ້ມູນທີ່ໄດ້ວ່າສັດປິດສະໜາຊະນິດນີ້ມີສາຍພັນເປັນ ສັດລົງລູກດ້ວຍນຳນິມ.

ຈະເຫັນວ່າ Decision Tree ແລະ Nearest Neighbor Classification ເປັນເຕັກນິກການຮຽນຮູ້ຂອງເຄື່ອງຈັກທີ່ແຕກຕ່າງກັນ, ໂດຍ Decision Tree ຈະເປັນວິທີການຈຳແນກເພື່ອຫາຄວາມຖືກຕ້ອງຂອງຂໍ້ມູນທີ່ສິນໃຈ, ໂດຍຮູ້ຄ່າຂອງຊຸດຂໍ້ມູນທັງໝົດ ເຊິ່ງເປັນໄປຕາມຫຼັກການຂອງ Supervised Learning ສ່ວນ, Nearest Neighbor ຈະພິຈາລະນາຈາກການທຳນາຍຄ່າຂອງຊຸດຂໍ້ມູນ ເຊິ່ງເປັນວິທີການແບບ Unsupervised Learning ໂດຍໃຫ້ຄວາມສໍາຄັນກັບຄວາມສໍາພັນຂອງຊຸດຂໍ້ມູນແທນການທຳນາວຄ່າຂອງຄວາມນໍາຈະເປັນຈາກຂໍ້ມູນທັງໝົດ. ດັ່ງນັ້ນ, ຈຶ່ງຕ້ອງພິຈາລະນາຊຸດຂໍ້ມູນທີ່ສິນໃຈວ່າຄວນເລືອກໃຊ້ວິທີການຈຳແນກກຸ່ມແບບໃດຈຶ່ງຈະເໝາະສົມທີ່ສຸດ.

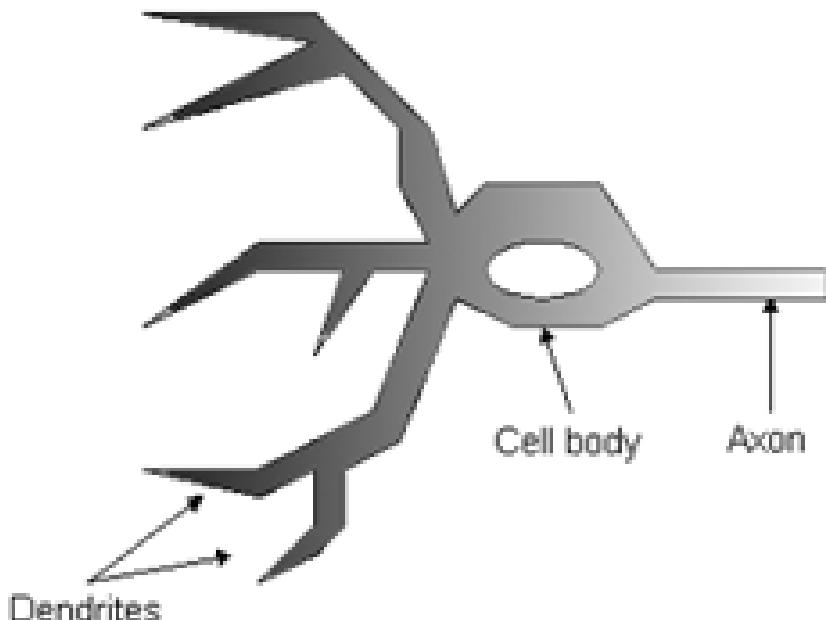
ଶବ୍ଦିକା

ການຮຽນຮູ້ຂອງເຄື່ອງຈັກ (Machine Learning) ໝາຍເຖິງເຕັກນິກ ຫຼື ຂະບວນການທີ່ໃຊ້ສໍາລັບປັບແຕ່ງ, ບຸງແຕ່ງເຄື່ອງຈັກ ຫຼື ຄອມພິວເຕີໃຫ້ມີພິດຕິກຳສະເພາະຕົວທີ່ສະໜັບສະໜຸນການຮຽນຮູ້ ແລະ ສຶກສາຂໍ້ມູນຈາກປະສົບການ. ເປັນວິທີທີ່ຊ່ວຍໃຫ້ເຄື່ອງຈັກສາມາດຮຽນຮູ້ໄດ້ຄືກັບມະນຸດ. ໂດຍຈະກຳວົຂອງກັບການພັດທະນາ ແລະ ອອກແບບອານຸກຳລິດທຶນ (Algorithm) ຫຼື ເຕັກນິກຕ່າງໆທີ່ຊ່ວຍໃຫ້ເຄື່ອງຈັກ ຫຼື ຄອມພິວເຕີມີຄວາມສາມາດໃນການຮຽນຮູ້ຂໍ້ມູນທີ່ສົນໃຈ ແລະ ເອົາໄປໃຊ້ໃນການແກ້ໄຂບັນຫາສະເພາະໜ້າໄດ້, ພ້ອມທັງເກັບສະສົມໄວ້ເປັນຄວາມຮູ້ ເຊິ່ງສາມາດເອົາໄປໃຊ້ໃນການແກ້ໄຂບັນຫາທີ່ອາດເກີດຂຶ້ນໄດ້ໃນອານາຄົດ. ການຮຽນຮູ້ສາມາດແບ່ງໄດ້ເປັນ 2 ຊະນິດຕື່: 1) Deductive ເປັນການຮຽນຮູ້ໂດຍອາໄສຄວາມຮູ້ທີ່ເປັນຈິງຢູ່ແລ້ວ ມີຄວາມຈິງເປັນສາກົນເຊິ່ງທຸກຄົນຍອມຮັບ, ໂດຍສາມາດຄາດການໄດ້ວ່າເຫດການດັ່ງກ່າວຈະເກີດຂຶ້ນແນ່ນອນຕາມຮູບແບບຂອງສິ່ງແວດລ້ອມ ແລະ 2) Inductive ເປັນການຮຽນຮູ້ຈາກເຫດການ ຫຼື ສິ່ງທີ່ສົນໃຈ, ໂດຍຮູ້ຂໍ້ມູນພຽງບາງສ່ວນແລ້ວຈຶ່ງເອົາໄປສຶກສາ ແລະ ສ້າງຄວາມເຂົ້າໃຈ ເພື່ອໄດ້ຂໍ້ມູນທັງໝົດຈົນກາຍເປັນຄວາມຈິງສາກົນທີ່ຄົນສ່ວນໝາຍຍອມຮັບ.

ບົດທີ 6 ເຄືອຂ່າຍເສັນປະສາດທຽມ

ເຄືອຂ່າຍເສັ້ນປະສາດທຸກມ ຫຼື ເຄືອຂ່າຍເສັ້ນປະສາດແບບຈຳລອງທາງຄະນິດສາດ ສໍາລັບການປະມວນຜົນຂໍ້ມູນດ້ວຍຄອມພິວເຕີທີ່ເນັ້ນໃສ່ການເຊື້ອມຕໍ່ (Connectionist) ເພື່ອຈຳລອງການເຮັດວຽກຂອງເຄືອຂ່າຍເສັ້ນປະສາດໃນສະໝອງມະນຸດ, ຈຸດປະສົງເພື່ອສ້າງເຄືອງມືທີ່ມີຄວາມສາມາດໃນການຮຽນຮູ້, ການຈິຈຳຮູບແບບ (Pattern Recognition) ແລະ ການຄັດຈຸອນຄວາມຮູ້ (Knowledge deduction) ຄືກັນກັບຄວາມສາມາດທີ່ມີໃນສະໝອງມະນຸດ.

ແນວຄົດທໍາອິດຂອງເຕັກນິກນີ້ໄດ້ມາຈາກການສຶກສາດ້ານເຄືອຂ່າຍໄຟຟ້າຊີ່ວະພາບ (Bioelectric Network) ໃນສະໝອງ, ເຊິ່ງປະກອບດ້ວຍ ແວລະບົບປະສາດ (Nerve Cells) ຫຼື “ນິວອອນ” (neurons) ແລະ ຈຸດເຊື່ອມຕໍ່ປະສາດ (Synapses), ແຕ່ລະເຊວປະສາດປະກອບດ້ວຍເສັ້ນເສັ້ນການຮັບກະແສປະສາດ ເອີ້ນວ່າ “ດົນໄດຣ໌” (Dendrite) ເຊິ່ງເປັນ Input ແລະ ເສັ້ນເສັ້ນການສົ່ງກະແສປະສາດເອີ້ນວ່າ “ແອກຊອນ” (Axon) ເຊິ່ງເປັນຕື່ Output ຂອງເຊວ ແວໜີ້ນີ້ເຮັດວຽກດ້ວຍການປະຕິກິລິຍາໄຟຟ້າເຄີມ ເມື່ອມີການກະຕຸນຈາກພາຍນອກ ຫຼື ກະຕຸນດ້ວຍເຊວນຳກັນ ກະແສປະສາດຈະແລ່ນຜ່ານດົນໄດຣ໌ເຂົ້າສູ່ແກ່ນນ້ອຍເຊິ່ງຈະເປັນຕົວຕັດສິນວ່າຕ້ອງກະຕຸນເຊວອື່ນໆຕໍ່ ຫຼືບໍ່? ຖ້າກະແສປະສາດແຮງພໍ ແກ່ນນ້ອຍກໍຈະກະຕຸນເຊວອື່ນໆຕໍ່ໄປຜ່ານທາງແອກຊອນຂອງມັນ. ແບບຈໍາລອງເຄືອຂ່າຍປະສາດເກີດຈາກການເຊື່ອມຕໍ່ລະຫວ່າງເຊວລະບົບປະສາດ ຈິນກາຍເປັນເຄືອຂ່າຍທີ່ເຮັດວຽກຮ່ວມກັນໄດ້



ກົບທີ 6.1 ສະແດງ Model ຂອງ Neuron ໃນສະໝັກອົງມະນຸດ

6.1 ទោរាមុខាយ

ລະບົບເຄືອຂ່າຍເສັ້ນປະສາດທຸກມ (Neural Network) ຫຼື “ເຄືອຂ່າຍເສັ້ນປະສາດທຸກມ (Artificial Neural Network: ANN)” ຂໍມາຍເຖິງຄອມພິວເຕີ້ຫີສາມາດຮຽນແບບການເຮັດວຽກຂອງສະໝອງມະນຸດ ດ້ວຍການປະມວນຜົນຂໍ້ມູນຂ່າວສານ ແລະ ຄວາມຮູ້ໄດ້ເກືອລະຫຼາຍໆ.

ນອກຈາກນີ້, ຍັງສາມາດຮັບ ແລະ ຈຶ່ງຈຳຂ່າວສານໃນຮູບແບບທີ່ເປັນປະສົບການໄດ້, ເຮັດໃຫ້ສາມາດ
ເຊື່ອມຕໍ່ຂີ້ເຫັດຈິງຫັງໝາຍເຂົ້າມຳກັນເພື່ອຫາຂໍ້ສະຫຼຸບ ແລະ ໄຊປະສົບການທີ່ຈັດເກັບໄວ້ມາຮຽນຮູ້ ແລະ
ສ້າງຄວາມເຂົ້າໃຈວ່າ ຂີ້ເຫັດຈິງໃໝ່ທີ່ໄດ້ຮັບເຂົ້າມາມີຄວາມກ່ຽວຂ້ອງກັນແນວໃດ ເພື່ອກໍາການປັບປຸງ
ຄວາມຮູ້ໃຫ້ມີຄວາມທັນສະໄໝເພື່ອປະໂຫຍດໃນອະນາຄິດ.

1. Neural Network เป็นตัวประมวลผลผ่านชั้นๆ ของข้อมูลให้ได้ ที่สำคัญขึ้นมาที่มีอยู่ คือ ความสามารถในการเรียนรู้ ปรับปรุง และตัดสินใจโดยอิสระ ไม่ต้องมีมนุษย์ดูแลอย่างต่อเนื่อง ทำให้สามารถประมวลผลข้อมูลในปริมาณที่มากและซับซ้อนได้ดี

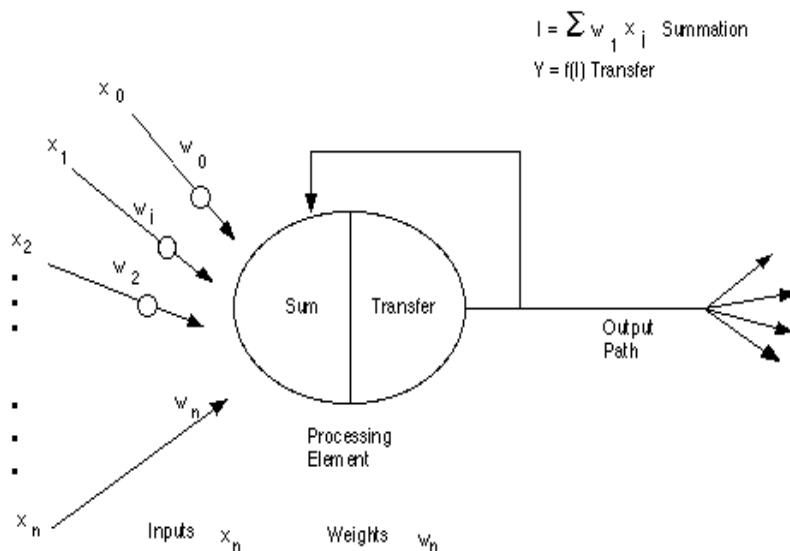
- សៀវភៅកប្បន្ន ត្រូវបានគេបង្ហាញ (Network) ដើម្បីបានចូលរួមទៅការងារ
 - ខ្លួនឯធម៌ និងសំណង់សំណង់ នឹងត្រូវបានគេបង្ហាញ (Synaptic) ដើម្បីបានចូលរួមទៅការងារ

2. เถือข่ายเส้นประสานทุ่ม และ เส้นประสานจิง

- ເຄືອຂ່າຍເສັ້ນປະສາດຫຼາມເປັນການຮຽນແບບການເຮັດວຽກຂອງສະໝອງມະນຸດ ທີ່ປະກອບໄປດ້ວຍ ແຂວົງເສດຖາງໜ້າຍທີ່ເຕັ້ນວ່າ “ເຊວລະບິບປະສາດ (Neuron)” ເຊິ່ງມີຫຼາຍກວ່າ 100 ຊະນິດ.
 - ເຊວປະສາດທີ່ມີຊະນິດດຽວກັນຈະຖືກຈັດໄວ້ໃນກຸ່ມດຽວກັນ ເຕັ້ນວ່າ “ເຄືອຂ່າຍ (Network)” ແຕ່ລະ ເຄືອຂ່າຍຈະບັນຈຸເຊວປະສາດຈຳນວນນັ້ນ 1000 ເຊວທີ່ມີການເຊື່ອມຕໍ່ກັນຢ່າງໝາງແໜ້ນ, ດັ່ງນັ້ນ ສະໝອງມະນຸດຈຶ່ງເຕັ້ນໄດ້ອີກຢ່າງໜຶ່ງວ່າ “ກຸ່ມປະສາດ”.
 - ການຮຽນແບບການເຮັດວຽກຂອງສະໝອງມະນຸດຂອງເຄືອງຄອມພິວເຕີ ເລີ່ມຈາກການກຳນົດໃຫ້ແຕ່ ລະຊອບແວເຕັ້ນວ່າ “ໂນດ (Node)” ຫຼາຍວ່າເປັນ “ເຊວລະບິບປະສາດ”
 - ແລະສ້າງການເຊື່ອມຕໍ່ໃຫ້ກັບໂນດເງື່ອນນັ້ນໃຫ້ເປັນເຄືອຂ່າຍ (Network), ແຕ່ລະເຄືອຂ່າຍຈະປະກອບໄປດ້ວຍໂນດທີ່ຖືກຈັດແບ່ງເປັນຊັ້ນງ່າງ ເຕັ້ນວ່າ “ເລເຍີ (Layer)” ແຕ່ລະເລເຍີຈະມີຫຼາຍທີ່ການເຮັດວຽກແຕກຕ່າງກັນ

6.2. ອົງປະກອບ ແລະ ໂຄງສ້າງການເຮັດວຽກ

Software ທີ່ຮູ່ນແບບເຄືອຂ່າຍຂອງເຊວປະສາດນັ້ນຈະມີຂອບເຂດ (Boundary) ຂັ້ນລະຫວ່າງຂໍ້ມູນນຳເຂົ້າກັບການເຮັດວຽກຂອງເສັ້ນປະສາດທຸກມີ, ເຊິ່ງປະກອບດ້ວຍເຄືອຂ່າຍຂອງເຊວທີ່ຖືກຈັດໄວ້ເປັນຂັ້ນ, ດັ່ງນັ້ນອົງປະກອບທີ່ໄດ້ຈັດແບ່ງເປັນຂັ້ນ ແລະ ໜ້າທີ່ຂອງແຕ່ລະອົງປະກອບຈຶ່ງມີດັ່ງນີ້



ຮູບທີ 6.2 ອົງປະກອບ ແລະ ໂຄງສ້າງການເຮັດວຽກຂອງເຄືອຂ່າຍເສັ້ນປະສາດທຸກມີ

6.2.1. ຂໍ້ມູນນຳເຂົ້າ (Input)

ຂໍ້ມູນນຳເຂົ້າຈະຖືກຈຳແນວກາຕາມຄຸນລັກສະນະ (Attribute) ແລ້ວ ຊັ້ນ: ຖ້າບັນຫາທີ່ລະບົບເສັ້ນປະສາດທຸກມີຈະຕ້ອງຕັດສິນໃຈຕື່ມີ ຕົວຢ່າງການອະນຸມັດເຖິງກົວ່າຈະໃຫ້ຜ່ານ ຫຼືບໍ່? ຂໍ້ມູນນຳເຂົ້າກຳຈະຖືກຈຳແນວກາເປັນຄຸນລັກສະນະຄື: ລະດັບລາຍຮັບ ແລະ ອາຍຸ ເປັນຕົ້ນ. ຂໍ້ມູນນຳເຂົ້ານອກຈາກຈະເປັນຂໍ້ຄວາມແລ້ວ ຍັງສາມາດເປັນຮູບພາບ ຫຼື ສົງກຳໄດ້, ແຕ່ອາດຈະຕ້ອງຜ່ານການແປງໃຫ້ເປັນສັນຍະລັກ ຫຼື ຕົວເລກເພື່ອໃຫ້ເຄື່ອງສາມາດສ້າງຄວາມເຂົ້າໃຈໄດ້ກ່ອນ ຈາກນັ້ນກຳຈະເຂົ້າສູ່ການເຮັດວຽກທີ່ແທ້ຈຶ່ງຂອງລະບົບເສັ້ນປະສາດທຸກມີທີ່ເລີ່ມຕົ້ນດ້ວຍການນຳຂໍ້ມູນເຂົ້າມາ, ໃຫ້ນໍ້າໜັກ (Weight) ຂອງຄວາມສຳພັນລະຫວ່າງຂໍ້ມູນນຳເຂົ້າເຫຼົ້ານັ້ນໃນຂັ້ນທຳອິດພາຍໃຕ້ຂອບເຂດຂອງລະບົບ.

6.2.2. ນັ້າໜັກ (Weight)

ເປັນສ່ວນປະກອບທີ່ສຳຄັນຂອງລະບົບເຄືອຂ່າຍເສັ້ນປະສາດ ເນື້ອງຈາກເປັນສ່ວນທີ່ໃຊ້ຫານໍ້າໜັກຂອງຄວາມສຳພັນລະຫວ່າງຂໍ້ມູນນຳເຂົ້າ, ວ່າຂໍ້ມູນນຳເຂົ້າໄດ້ມີຄວາມສຳພັນກັບຂໍ້ມູນນຳເຂົ້າອື່ນໃນລະດັບ ໄດ້, ເຊິ່ງຈະຮັດໃຫ້ສາມາດເຊື່ອມຕໍ່ໄປທາຂໍ້ສະຫຼຸບໄດ້ ດ້ວຍການລອງຜິດລອງຖືກໃນຄວາມສຳພັນແຕ່ລະແບບ ແລະ ເກັບໄວ້ເປັນແບບແຜນ ຫຼື ຮູບແບບ (Pattern) ຂອງປະສົບການເພື່ອການຮຽນຮູ້ຂອງເຄືອຂ່າຍ.

6.2.3. ຜົງຊັ້ນການລວມ (Summation Function)

เป็นเดิอช่ายที่รักษาที่ในงานล้อมค่าม้าหัวที่ได้จากเดิอช่ายในขั้น input ដ้ือสະຫຼຸບ
ຜົນຄວາມສໍາພັນລະຫວ່າງຂໍ້ມູນນຳເຊົ້າ ລົກງານແປງເປັນຂໍ້ມູນຂ່າວສານທີ່ມີຄວາມໝາຍໃນຂັ້ນຕໍ່ໄປ.

6.2.4. ພັງຊັນການແປງ (Transformation Function)

เป็นเต็وخ່າຍທີ່ເຮັດໜ້າທີ່ໃນການເຊື່ອມຕໍ່ (Integrate) ຂ່າວສານທີ່ຜ່ານການປະມວນຜົນຈາກເຄື່ອຂ່າຍໃນຊັ້ນຕ່າງໆ, ແລ້ວກໍ່ແປງ (Transform) ໃຫ້ກາຍເປັນຂ່າວສານທີ່ສື່ຄວາມໝາຍ ແລະ ເປັນປະໂຫຍດຕໍ່ການເອົາໄປໃຊ້ໄດ້ເພື່ອສົ່ງອອກໄປເປັນຜົນຮັບ (Output).

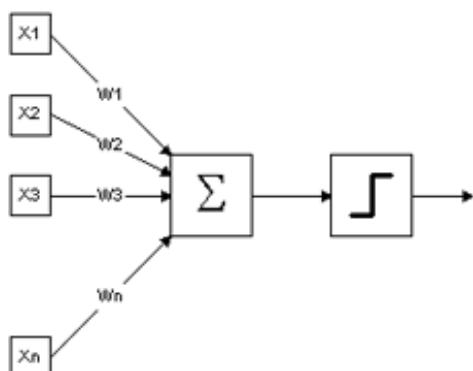
6.2.5. ຜົນຮັບ (Output)

ຜົນຮັບທີ່ໄດ້ຈາກເຄືອຂ່າຍເສັ້ນປະສາດຖຸມ ຈະໝາຍເຖິງແນວທາງໃນການແກ້ໄຂບັນຫາ, ເຊັ່ນ:
ບັນຫາການອະນຸມັດເງິນກູ້ວ່າຜູ້ກູ້ຈະຜ່ານການອະນຸມັດ ຫຼືບໍ່? “ຜົນຮັບ” ທີ່ຜູ້ໃຊ້ຈະໄດ້ຮັບຄື “ອະນຸມັດ”
ຫຼື “ບໍ່ອະນຸມັດ”, ແຊັ່ງເຄືອຂ່າຍເສັ້ນປະສາດທຸມຈະໃຊ້ສັນຍະລັກແໜນຄຳຕອບທັງໝົດ.

ຜົນຮັບທີ່ໄດ້ຈາກເຄືອຂ່າຍໜຶ່ງສາມາດເປັນຂໍ້ມູນນຳເຂົາ (Input) ຂອງເຄືອຂ່າຍໜຶ່ງໄດ້, ໃນນີ້ເພື່ອເປັນຂໍ້ມູນນຳເຂົາຂອງການຕັດສິນໃຈແກ້ໄຂບັນຫາອື່ນ ເຊັ່ນ: ຜົນຮັບທີ່ໄດ້ຈາກການອະນຸມັດເງິນກູ້ ອາດຈະນຳໄປໃຊ້ເປັນຂໍ້ມູນນຳເຂົາເພື່ອການອະນຸມັດສິນເຊື້ອທີ່ຢູ່ອາໄສໄດ້.

6.3. ໂຄງສໍາ

ນັກວິໄຈສ່ວນໝາຍໃນປັດຈຸບັນເຫັນກົງກັນວ່າເຄືອຂ່າຍເສັ້ນປະສາດທຸງມີໂຄງສ້າງແຕກຕ່າງຈາກເຄືອຂ່າຍເສັ້ນສະໜອງມະນຸດ ແຕ່ກໍຍັງຄືສະໜອງ ໃນທາງທີ່ວ່າເຄືອຂ່າຍເສັ້ນປະສາດທຸງ ຄືການລວມກຸ່ມແບບຂະໜານຂອງໜ່ວຍປະມວນຜົນຍ່ອຍໆ ແລະ ການເຊື່ອມຕໍ່ນີ້ເປັນສ່ວນສຳຄັນທີ່ເຮັດໄຫ້ເກີດສະຕິປັນຍາຂອງເຄືອຂ່າຍ. ເມື່ອພິຈາລະນາຂະໜາດແລ້ວ ສະໜອງມະນຸດມີຂະໜາດໃຫຍ່ກ່ວ່າເຄືອຂ່າຍເສັ້ນປະສາດທຸງໝາຍ ລວມທັງເຊວປະສາດຍັງມີຄວາມຊັບຊອນກວ່າໜ່ວຍຢ່ອຍຂອງເຄືອຂ່າຍ ຢ່າງໃດກໍດີໜ້າທີ່ສຳຄັນຂອງສະໜອງ ເຊັ່ນການຮຽນຮູ້ຍັງສາມາດຖືກຈຳລອງຂຶ້ນຢ່າງຍໍດ້ວຍເຄືອຂ່າຍເສັ້ນປະສາດທຸງນີ້.



ຮບທີ 6.3 ສະແດງ Model ຂອງ Neuron ໃນຄອມພິວເຕີ

6.4. ប្រព័ន្ធបាតា

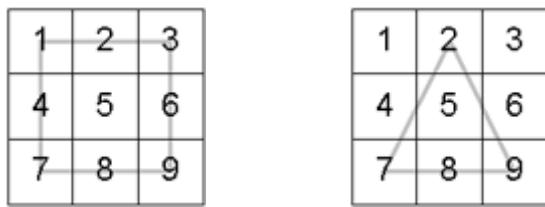
สำลับในคอมพิวเตอร์ Neurons ประกอบด้วย input และ output ตัวรับ โดยจำลองให้ input แต่ละอันมี weight เป็นตัวกำหนดน้ำหนักของ input โดย neuron แต่ละตัวจะมีค่า threshold เป็นตัวกำหนดว่ามีน้ำหนักกี่อย่าง input ต้องมากเท่าใดจึงจะสามารถส่ง output ไปที่ neurons ตัวอื่นได้ เมื่อเอ้า neuron แต่ละตัวจะมาร่วมกัน ภาระที่ต้องรับจะถูกแบ่งเป็นส่วนๆ ตามจำนวน neuron ที่มีอยู่ แต่ละตัวจะคำนวณผลรวมของ input ที่ได้รับแล้วนำค่าไปหารด้วยค่า threshold ที่ตั้งไว้ ถ้าผลหารน้อยกว่า threshold จะไม่ส่ง output ไปยัง neuron ตัวอื่น แต่ถ้าผลหารมากกว่า threshold จะส่ง output ไปยัง neuron ตัวอื่น

6.5. ການເຮັດວຽກ

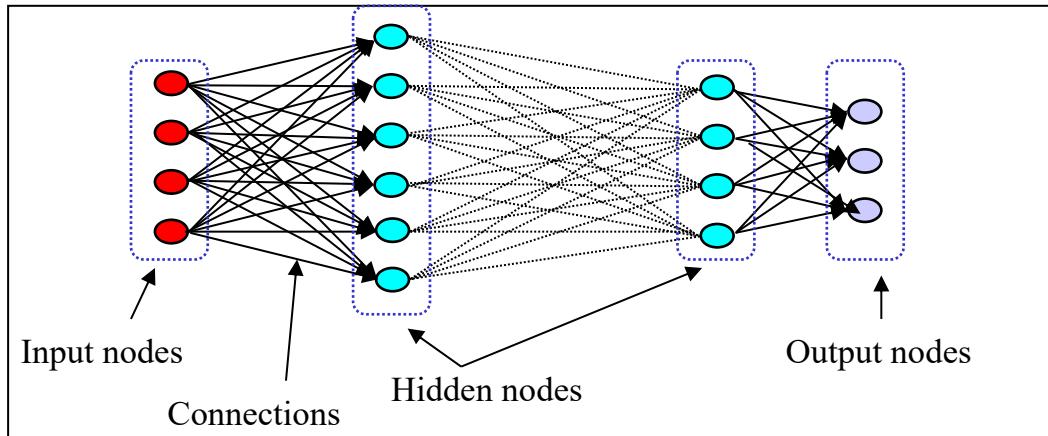
ການເຮັດວຽກຂອງ Neural Networks ແມ່ນເມື່ອມີ input ເຂົ້າມາທີ່ network, network ກໍາເອົາ input ມາຄຸນກັບ weight ຂອງແຕ່ລະຂາ, ຜົນທີ່ໄດ້ຈາກ input ຫຼັກງານຂາຂອງ neuron ຈະເອົາມາລວມກັນແລ້ວກໍເອົາມາຫຼູບກັບ threshold ທີ່ກໍານົດໄວ້. ຖ້າຜົນທັງໝົດມີຄ່າຫຼາຍກວ່າ threshold ແລ້ວ neuron ກໍຈະສົ່ງ output ອອກໄປ, output ນີ້ກໍຈະຖືກສົ່ງໄປຢ້າງ input ຂອງ neuron ອື່ນໆ ທີ່ເຊື່ອມກັນໃນ network. ຖ້າຄ່ານັ້ນອຍກວ່າ threshold ກໍຈະບໍ່ເກີດ output. ການເຮັດວຽກຂອງ neural ສາມາດຂົງອອກມາໄດ້ດັ່ງນີ້:

if (sum(input * weight) > threshold) then output

ສິ່ງສຳຄັນຄືເຮົາຕ້ອງຮູ້ຄ່າ weight ແລະ threshold ສໍາລັບສິ່ງທີ່ເຮົາຕ້ອງການເພື່ອໃຫ້ຄອມພົວເຕີຮູ້, ເຊິ່ງເປັນຄ່າທີ່ບໍ່ແມ່ນອນ ແຕ່ສາມາດກຳນົດໃຫ້ຄອມພົວເຕີປັບຄ່າເຫຼົ່ານັ້ນໄດ້ ໂດຍການສອນໃຫ້ມັນຮູ້ຈັກ pattern ຂອງສິ່ງທີ່ເຮົາຕ້ອງການໃຫ້ມັນຈີ່ຈຳ ເອີ້ນວ່າ "back propagation" ເຊິ່ງເປັນຂະບວນການກັບຄືນຂອງການຈີ່ຈຳ. ໃນການຝຶກ feed-forward neural networks ຈະມີການໃຊ້ອານຸກົດທີ່ມີແບບ back-propagation ເພື່ອໃຊ້ໃນການປັບປຸງນຳໜັກຄະແນນຂອງເຄືອຂ່າຍ (network weight), ຫຼັງຈາກໄສຮູບແບບຂໍ້ມູນສໍາລັບຝຶກໃຫ້ແກ່ເຄືອຂ່າຍໃນແຕ່ລະເທື່ອແລ້ວ ຄ່າທີ່ໄດ້ຮັບ (output) ຈາກເຄືອຂ່າຍຈະຖືກນຳໄປປູງທຸງບວກຜົນທີ່ຄາດຫວັງ, ແລ້ວກໍສ້າງການຄິດໄລ່ຫາຄ່າຄວາມຜິດພາດ, ເຊິ່ງຄ່າຄວາມຜິດພາດນີ້ຈະຖືກສົ່ງກັບເຂົ້າສູ່ເຄືອຂ່າຍເພື່ອໃຊ້ແກ້ໄຂຄ່ານຳໜັກຄະແນນຕໍ່ໄປ.



ຮັບທີ 6.4 ສະແດງການແຍກລະຫວ່າງສິ້ງມ ແລະ ສາມຫຼູມ



ຮັບທີ 6.5 ສະແດງໄຕງສ້າງວິຈອນເຄືອຂອງ Neural Network

❖ Output ຂອງແຕ່ລະ Node

$$y_i = f(w_i^1 x_1 + w_i^2 x_2 + w_i^3 x_3 + \dots + w_i^m x_m)$$

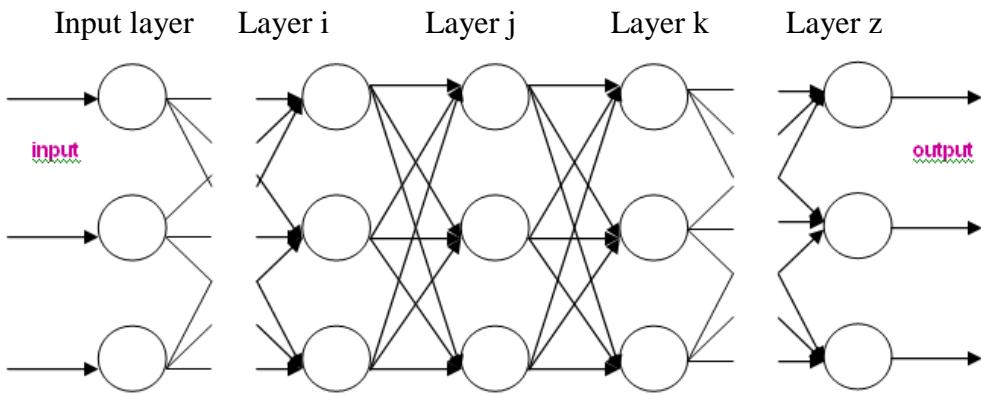
$$= f(\sum_j w_i^j x_j)$$

ເມື່ອ X_i = input ຈາກໂນດອິນໆ

Wij = ນັ້ງຫັກ (weight) ຂອງຕົວລະແຂນ (connection)

6.5.1. Back propagation Algorithm

Back-propagation เป็นอานวัติเดิมที่ใช้ในการเรียนรู้ของเครือข่ายเส้นประสาดวิธีนี้ที่นิยมใช้ใน Multilayer Perceptron เพื่อปับค่าบัน្តោយការในเส้นเชื่อมต่อระหว่างโนดให้เข้าមាមសិរី โดยការបែបគាំនីមួយៗនឹងរាបតាមរាងចក្រការណ៍ការពារនៃការបែបគាំនីមួយៗដែលត្រូវបានរាយការណ៍ឡើង។ ដូច្នេះការបែបគាំនីមួយៗនឹងរាបតាមរាងចក្រការណ៍ការពារនៃការបែបគាំនីមួយៗដែលត្រូវបានរាយការណ៍ឡើង។



រូបទី 6.6 សម្រាប់នូវបច្ចាស់នៃការគ្រប់គ្រងនៃតម្លៃនៃការបញ្ចូល (Back-propagation)

❖ ខ្លួនខោះ Back-propagation Algorithm មិត្តឱ្យនឹង

1. រាយមិត្តតាមតម្លៃនៃការបញ្ចូល (Rate Parameter : r)
 2. សំឡុបពេលពិវិះគ្នា input ឲ្យបញ្ចូលខ្លួនព័ត៌មានថ្មីជាការវាយតម្លៃនៃការបញ្ចូល
- តិចតាមតាមតម្លៃនៃការបញ្ចូលដើម្បីបញ្ចូលពិវិះគ្នាដែលត្រូវបានស្វែងរក
 - តិចតាមតាមតម្លៃនៃការបញ្ចូលពិវិះគ្នាដែលត្រូវបានស្វែងរក
 - ឱ្យបញ្ចូលខ្លួន (Output Layer)

$$\beta_z = d_z - o_z$$

មើល d_z ແມ່ນតាមតម្លៃនៃការបញ្ចូល

Oz ແມ່ນតាមតម្លៃនៃការបញ្ចូល

- ឱ្យបញ្ចូលខ្លួន (Hidden Layer)

$$\beta_j = \sum_{k=1}^n w_{jk} * o_k * (1 - o_k) * \beta_k$$

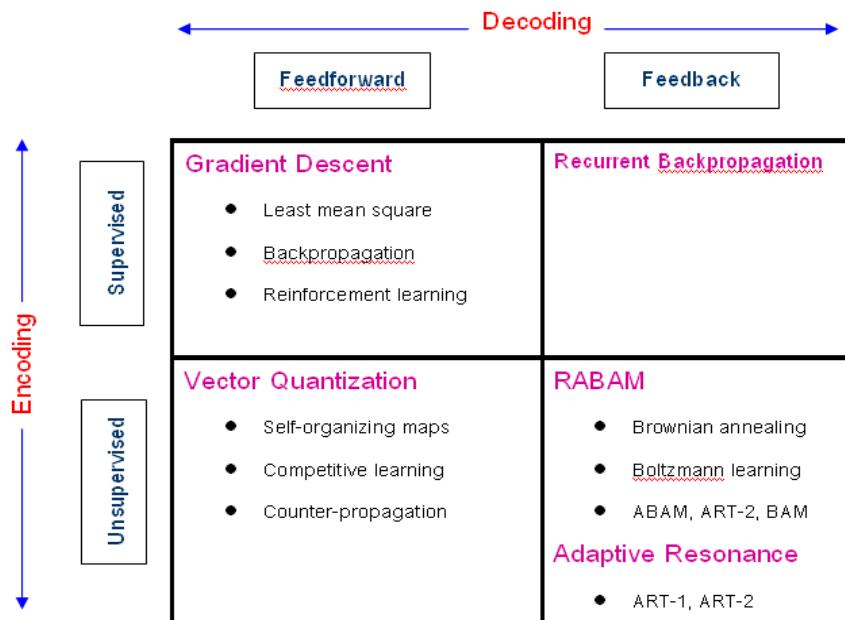
មើល w_{jk} ແມ່ນប៉ាន្ទីរកខ្លួនដើម្បីបញ្ចូលខ្លួនព័ត៌មានតាមតម្លៃនៃការបញ្ចូល

- តិចតាមតាមតម្លៃនៃការបញ្ចូលពិវិះគ្នាដែលត្រូវបានស្វែងរក ដោយសិរិចិនព័ត៌មាន

$$\Delta w_{ij} = r * o_i * o_j * (1 - o_j) * \beta_j$$

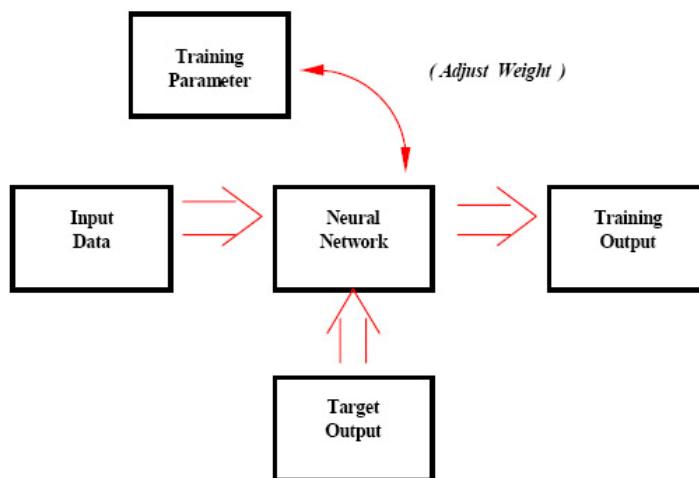
- ផើមតាមតម្លៃនៃការបញ្ចូលពិវិះគ្នាដែលត្រូវបានស្វែងរក សំឡុបពិវិះគ្នា input ហើយ និង និង ប៉ាន្ទីរកខ្លួន

6.5.2. ការរួមទឹកសំគាល់បែង Neural Network



1. ការរួមទឹកបែងមិភាពសោរ (Supervised Learning)

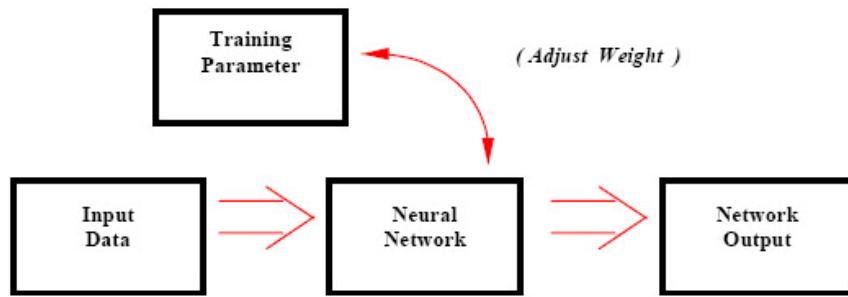
បែងការរួមទឹកបែងមិភាពភាពតាំពេលដើម្បីខ្សោយវិធាននៃការសោរ។ ផ្តល់ខ្លួនអ្នកសោរ ឬវិធាននៃការសោរដូចជាការបង្កើតការងារ ឬការសោរដូចជាការសោរដែលត្រូវបានបង្កើតឡើង។ តាមការសោរបានបង្កើតឡើង ត្រូវបានបង្កើតឡើងដោយការសោរដែលត្រូវបានបង្កើតឡើង។ (ប្រព័ន្ធបានបង្កើតឡើងដោយការសោរដែលត្រូវបានបង្កើតឡើង)។



រូបទី 6.7 សម្រាប់ការរួមទឹកបែងមិភាពសោរ (Supervised Learning)

2. ការរួមទឹកបំមិភាពសោរ (Unsupervised Learning)

បែងការរួមទឹកបំមិធ្យុយណ៍នាំ, បំមិភាពភាពតាំពេលវា ឬដី, ឬវិធាននៃការសោរដូចជាការសោរដែលត្រូវបានបង្កើតឡើង។ ជាមួយការសោរដែលត្រូវបានបង្កើតឡើង ដូចជាការសោរដែលត្រូវបានបង្កើតឡើង។

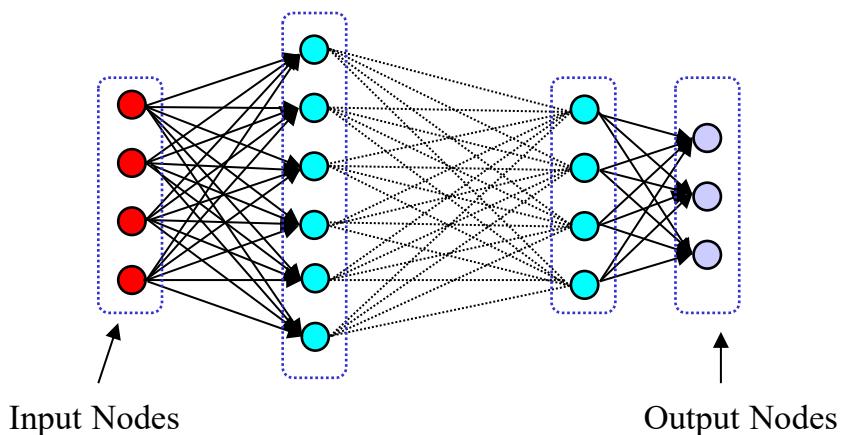


ຮູບທີ 6.8 ສະແດງການຮຽນຮູ້ແບບບໍ່ມີການສອນ Unsupervised Learning

6.6. Network Architecture

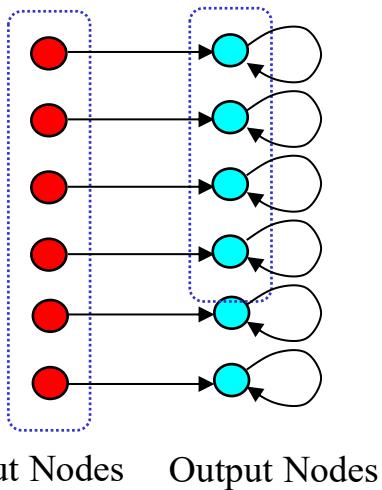
6.6.1. Feedforward network

ຂໍ້ມູນທີ່ປະມວນຜົນໃນວົງຈອນເລື້ອຂ່າຍຈະຖືກສິ່ງໄປໃນຫຼັດທາງດຽວຈາກ Input Nodes ສິ່ງຕໍ່ມາເລື້ອຍໆ ຈຶນເຖິງ Output Nodes ໂດຍບໍ່ມີການກັບຄືນຂອງຂໍ້ມູນ ຫຼື ລວມທັງ Nodes ໃນ layer ດຽວກັນກຳບໍ່ມີການເຊື່ອມຕໍ່ກັນ.



ຮູບທີ 6.9 ສະແດງສະຖາປັດຕະຍະກຳຂອງ Feedforward network

6.6.2. Feedback network



ຮູບທີ 6.10 ສະແດງສະຖາບັດຕະຍະກຳຂອງ Feedback network

6.6.3. Network Layer

ພື້ນຖານທີ່ສໍາຄັນຂອງ Artificial Neural Network ປະກອບໄປດ້ວຍ 3 ສ່ວນ ຫຼື 3 layer ໄດ້ແກ່ ຂັ້ນຂອງ input units ທີ່ຖືກເຊື່ອມຕໍ່ກັບຂັ້ນຂອງ hidden units ເຊິ່ງເຊື່ອມຕໍ່ກັບຂັ້ນຂອງ output units

- ການເຮັດວຽກຂອງ input unit ຈະເຮັດໜ້າທີ່ແກນສ່ວນຂອງຂໍ້ມູນດີບ ທີ່ຈະຖືກບ້ອນເຂົ້າສູ່ເລືອຂ່າຍ
- ການເຮັດວຽກຂອງແຕ່ລະ hidden units ຈະຖືກກຳນົດ ໂດຍການເຮັດວຽກຂອງ input units ແລະ ຄ່ານັ້ນກາເຫິງຄວາມສໍາພັນລະຫວ່າງ input units ແລະ hidden units
- ພິດຕິກຳການເຮັດວຽກຂອງ output units ຈະຂຶ້ນຢູ່ກັບການເຮັດວຽກຂອງ hidden units ແລະ ຄ່ານັ້ນກາລະຫວ່າງ hidden units ແລະ output units

❖ Architecture of Layer

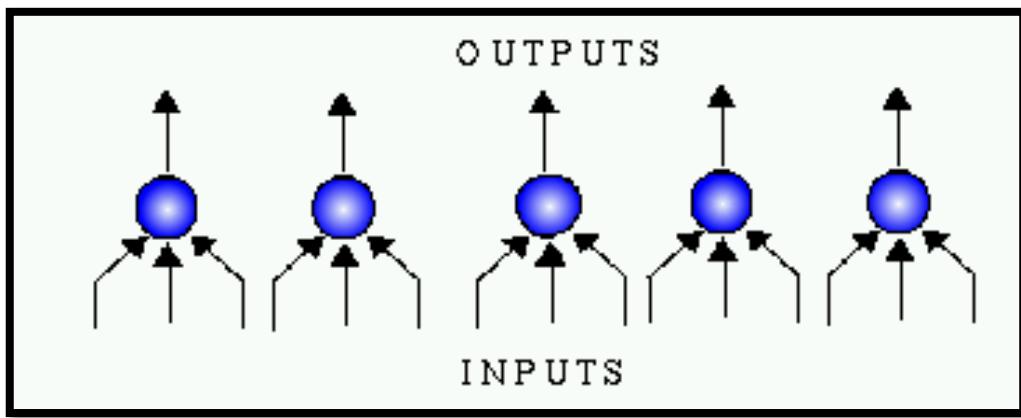
ສາມາດຈຳແນກສະຖາບັດຕະຍະກຳຂອງຂັ້ນ (layer) ອອກເປັນ 2 ປະເພດຕີ Single-layer ແລະ Multi-layer

- **Single-layer perceptron** ເຄືອຂ່າຍເສັ້ນປະສາດທີ່ປະກອບດ້ວຍຂັ້ນພຽງຂັ້ນດູວ ຈຳນວນ input nodes ຂຶ້ນຢູ່ກັບຈຳນວນ components ຂອງ input data ແລະ Activation Function ຂຶ້ນຢູ່ກັບລັກສະນະຂໍ້ມູນຂອງ Output ເຊັ່ນ: ຖ້າ output ທີ່ຕ້ອງການເປັນ “ແມ່ນ” ຫຼື “ບໍ່ແມ່ນ” ເຮົາຈະຕ້ອງໃຊ້ Threshold function

$$f(x) = \begin{cases} 1 & \text{if } x \geq T \\ 0 & \text{if } x < T \end{cases} \quad T \text{ ແມ່ນ Threshold level}$$

ບື້ຖ້າ output ເປັນຄ່າຕົວເລກທີ່ຕໍ່ເນື້ອງ ເຮົາຕ້ອງໃຊ້ continuous function ເຊັ່ນ Sigmoid function

$$f(x) = \frac{1}{1 + e^{-\alpha x}}$$



ຮບທີ 6.11 ສະແດງ Single-layer perceptron

- **Multi-layer perceptron** ເຄືອຂ່າຍເສັ້ນປະສາດຈະປະກອບດ້ວຍໝາຍຊັ້ນໂດຍໃນແຕ່ລະຊັ້ນຈະປະກອບດ້ວຍໂນດ (nodes) ຫຼື ຖົງບໄດ້ກັບຕົວເຊວລະບົບປະສາດ(neurons) ຄ່ານຳໜັກຂອງເສັ້ນທີ່ເຊື່ອມຕໍ່ລະຫວ່າງໂນດຂອງແຕ່ລະຊັ້ນ (ມາຕຣິດ W), ຄ່າ bias vector (b) ແລະ ຄ່າ output vector (a) ໂດຍ m ເປັນຕົວເລກບອກລຳດັບຊັ້ນກຳກັບໄວ້ດ້ານເທິງ ເມື່ອ p ເປັນ input vector ການຄົດໄລ່ຄ່ານຳເຂົ້າສຳລັບເຄືອຂ່າຍເສັ້ນປະສາດທີ່ມີ M ຊັ້ນຈະເປັນດັ່ງສິນຜົນ

$$a^{m+1} = f^{m+1} * (W^{m+1} * a^m + b^{m+1})$$

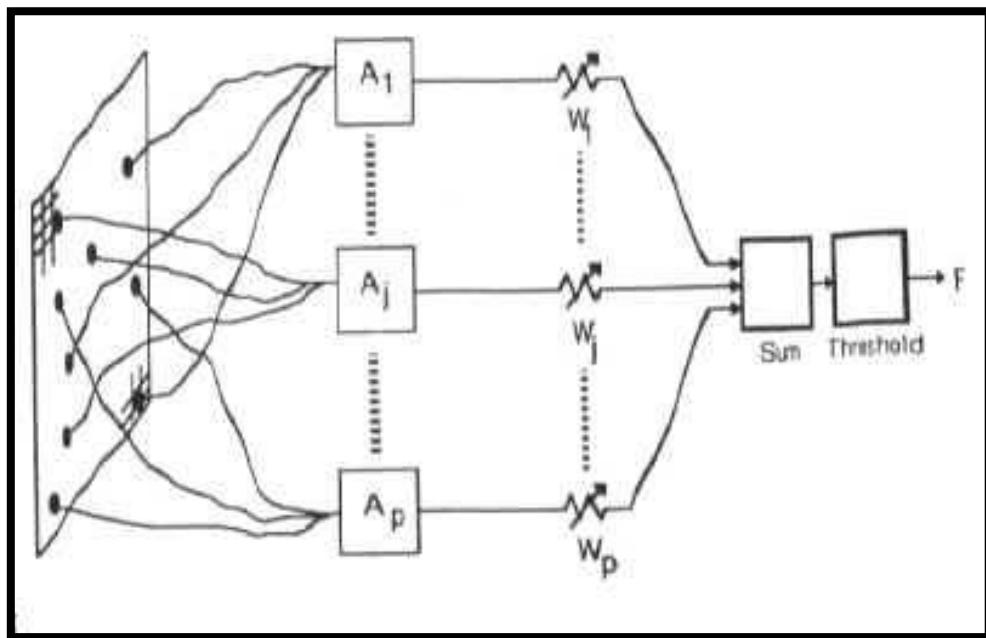
ເມື່ອ m ມີຄ່າ 0, 2, ..., M-1

a^0 ແມ່ນ p

a ແມ່ນ a^m ແລະ f ເປັນ transfer function

6.6.4. Perceptrons

ໃນຍຸກສັດຕະວັດ 60 ວົງກສ່ວນໃຫຍ່ຂອງເຄືອຂ່າຍໄດ້ຮັບການວິພາກວິຈານ ໃນຫົວຂໍເລື່ອງ Perceptrons ເຊິ່ງຄົ້ນພົບໂດຍ Frank Rosenblatt, perceptron ກາຍເປັນ MCP model (Neuron with Weighted Inputs) ພ້ອມກັບສ່ວນເພີ່ມເຕີມ. ຈາກຮູບທີ 6.12 ໃນສ່ວນຂອງ A1, A2, Aj, Ap ເຮັດວຽກເພື່ອຄັດເລືອກສິ່ງທີ່ແຕກຕ່າງອອກມາຈາກຮູບພາບທີ່ຮັບເຂົ້າໄປໂດຍ perceptrons ສາມາດສຳເນົາຄວາມຄົດພື້ນຖານພາຍໃນຂອງສັດລົງງລູກດ້ວຍນິມ ຫຼັກງານແລ້ວຈະໃຊ້ໃນຮູບແບບ recognition ແລະ ສາມາດຂະຫຍາຍໃຫ້ມີຄວາມສາມາດສູງກວ່ານີ້.



ຮັບທີ 6.12 ສະແດງໄລ່ງສ້າງຂອງ Perceptrons

6.7. ປະໂຫຍດຂອງເຄືອຂ່າຍເສັ້ນປະສາດທາມ

1. ເກີດຂໍ້ຜິດພາດໄດ້ຢາກ (Fault Tolerance)

ໜາກລະບົບເຄືອຂ່າຍເສັ້ນປະສາດທຸງປະກອບໄປດ້ວຍເຄືອຂ່າຍທີ່ໃຊ້ໃນການປະມວນຜົນຫຼາກຫຼາຍເຄືອຂ່າຍ, ຄວາມຜິດພາດທີ່ເກີດຂຶ້ນຈາກພູງໜຶ່ງ ຫຼືສອງເຄືອຂ່າຍຈະບໍ່ເຮັດໃຫ້ລະບົບທັງໝົດເກີດຂໍຜິດພາດໄດ້.

2. ទារាមសាមាតិនការងារមាត្រាបេណជិំ (Generalization)

ເມື່ອລະບົບເຕີອຂ່າຍເສັ້ນປະສາດທຽມໄດ້ຮັບຂໍ້ມູນນຳເຂົາທີ່ບໍ່ຄົບຖວນ ຫຼື ບໍ່ພຽງພໍຕໍ່ການຫາຂໍ້ສະຫຼຸບ, ຫຼື ໄດ້ຮັບຂໍ້ເທັດຈິງທີ່ບໍ່ເຄີຍໄດ້ຮັບມາກ່ອນ, ລະບົບຈະສາມາດລຳດັບການເຊື່ອມຕໍ່ຂໍ້ເທັດຈິງຈົນສາມາດໃຫ້ຂໍ້ສະຫຼຸບ ແລະ ເຫດຜົນໄດ້.

3. ຄວາມສາມາດໃນການປັບປ່ງ (Adaptability)

ເຄືອຂ່າຍເສັ້ນປະສາດຖຽມສາມາດຮົງຮູ້ສະພາບແວດລ້ອມໃໝ່ໄດ້, ດ້ວຍນັ້ນເມື່ອມີເຫດການໃໝ່ງໆເຊົາສ່ລະບົບ, ລະບົບກຳຈະສາມາດບັບປ່ຽນ ຫຼື ປັບປຸງຄວາມຮັກໃຫ້ທັນສະໄໝຕາມເຫດການໃໝ່ນັ້ນ.

4. ความสามารถในการพยากรณ์ (Forecasting Capability)

ເຄືອຂ່າຍເສັ້ນປະສາດທຸກມສາມາດນຳຂຶ້ນທາງສະຖິຕີເກົ່າທີ່ມີຢູ່ໃນລະບົບເອົາມາໃຊ້ໃນການຄາດຄະເນ ຫຼື ພະຍາກອນຂຶ້ນຢູ່ໃນອະນາຄົດໄດ້.

6.8 ການປະຍຸກໃຊ້ Neural Network

ແບບເຄືອຂ່າຍລະບົບປະສາດ (Neural Network) ເນື້ອງຈາກຄວາມສາມາດໃນການຈຳລອງພິດຕີກຳທາງກາຍະພາບຂອງລະບົບທີ່ມີຄວາມຊັບຊັອນຈາກຂຶ້ນນີ້ປ້ອນໃຫ້ຮູນຮູ້, ການປະຍຸກໃຊ້ເຄືອ

ຂ່າຍລະບົບປະສາດຈຶ່ງເປັນທາງເລືອກໃໝ່ໃນການຄວບຄຸມ, ເຊິ່ງມີຜູ້ນໍາມາປະຢູກໃຊ້ວຽກຫຼາຍປະເພດໄດ້ແກ່:

1. ວຽກການຈີ່ຈໍາຮູບແບບທີ່ມີຄວາມບໍ່ແນ່ນອນ ເຊັ່ນ: ລາຍມື, ລາຍເຊັນ, ຕົວອັກສອນ ແລະ ຮູບໜ້າ
2. ວຽກການປະມານຄ່າຟັງຊັນ ຫຼື ການປະມານຄວາມສໍາພັນ (ມີ inputs ແລະ outputs ແຕ່ບໍ່ຮູ້ວ່າ inputs ກັບ outputs ມີຄວາມສໍາພັນກັນຢ່າງໃດ).
3. ວຽກທີ່ມີສິ່ງແວດລ້ອມປ່ຽນແປງຢູ່ສະເໜີ (ວົງຈອນເຄືອຂ່າຍນິວອອນສາມາດປັບຕົວເອງໄດ້).
4. ວຽກຈັດໝວດໝູ່ ແລະ ແຍກສິ່ງຂອງ.
5. ວຽກພະຍາກອນ ເຊັ່ນ: ພະຍາກອນອາກາດ, ພະຍາກອນຫຼຸ້ນ...
6. ການປະຢູກໃຊ້ເຄືອຂ່າຍລະບົບປະສາດຄວບຄຸມຂະບວນການທາງເຄມີໂດຍວິທີພະຍາກອນແບບຈຳລອງ (Model Predictive Control)
7. ການປະຢູກໃຊ້ເຄືອຂ່າຍລະບົບປະສາດແບບແພ່ກະຈາຍກັບ ໃນການພະຍາກອນພະລັງງານຄວາມຮອນທີ່ສະສົມຢູ່ໃນຕີກອາຄານ
8. ການໃຊ້ເຄືອຂ່າຍລະບົບປະສາດໃນການທ່າ psychometric chart, ການປະຢູກໃຊ້ເຄືອຂ່າຍລະບົບປະສາດຄວບຄຸມລະບົບ HVAC.

6.8.1 ການປະຢູກໃຊ້ເຄືອຂ່າຍເສັ້ນປະສາດທຸງມີໃນວຽກຫຼາຍກິດ

- ການເຮັດເໝີອງຂໍ້ມູນ (Data Mining) ເປັນການເພີ່ມຄວາມສາມາດໃນການຄົ້ນຫາຂໍ້ມູນໃນຖານຂໍ້ມູນຕ່າງຊະນິດ ຫຼື ຖານຂໍ້ມູນທີ່ມີຂະໜາດໃຫຍ່ ແລະ ຊັບຊັອນໄດ້.
- ການປ້ອງກັນການໂກງພາສີ (Tax Fraud) ຂ່ວຍລະບຸ ແລະ ຄົ້ນຫາການເຮັດວຽກທີ່ຜິດກິດໝາຍໃນດ້ານການເສຍພາສີໄດ້.
- ການບໍລິການທາງດ້ານການເງິນ (Financial Service) ຂ່ວຍພັດທະນາຮູບແບບການບໍລິການທາງດ້ານການເງິນ ເຊັ່ນ: ການໃຫ້ຂໍ້ມູນຕະຫຼາດຫຼຸ້ນ ແລະ ເປັນຜູ້ຂ່ວຍການຄ້າຫຼຸ້ນ ເປັນຕົ້ນ.
- ການວິເຄາະຜະລິດຕະພັນໃໝ່ (New Product Analysis) ຂ່ວຍພະຍາກອນຍອດຂາຍ ແລະ ເລືອກຕະຫຼາດກຸ່ມເປົ້າໝາຍໄດ້.
- ການຈັດການຄ່າທຳນຽມສາຍການບິນ (Airline Fare Management) ຂ່ວຍພະຍາກອນປະລິມານຄວາມຕ້ອງການໃນການຈອງປິ່ຍິນ ແລະ ຈັດຕາຕະລາງກຳລັງຄົນໄດ້.
- ການປະເມີນຜົນ ແລະ ຄັດເລືອກພະນັກງານໃໝ່ ຂ່ວຍຄັດເລືອກພະນັກງານໃໝ່ທີ່ມີຄຸນສົມບັດຕາມທີ່ອີງກອນຕ້ອງການໄດ້.
- ຈັດສັນຊັບພະຍາກອນໃນອີງກອນໂດຍອາໄສຂໍ້ມູນໃນອາດີດ, ເຄືອຂ່າຍເສັ້ນປະສາດທຸງມີຈະຊ່ວຍສັນຊັບພະຍາກອນທັງໝົດໃນອີງກອນ, ໂດຍອາໄສຂໍ້ມູນໃນອາດີດ ແລະ ທົດລອງປ່ຽນຄ່າຂໍ້ມູນເພື່ອໃຫ້ໄດ້ຜົນຕອບແກນສູງສຸດ.
- ກວດສອບລາຍເຊັນ (Signature Validation) ຂ່ວຍໃນການກວດສອບລາຍເຊັນຈົງກັບລາຍເຊັນທີ່ຈັດເກັບໄວ້ໃນແພັນຂໍ້ມູນ

6.8.2. ຕົວຢ່າງການເຮັດວຽກຂອງເຄືອຂ່າຍເສັ້ນປະສາດ

ເຕັກໂນໂລຢີເຄືອຂ່າຍເສັ້ນປະສາດຈັດວ່າເປັນເຕັກໂນໂລຢີທີ່ມີຄວາມສາມາດສູງ ຈຶ່ງໄດ້ມີການນຳໄປປະຍຸກໃຊ້ກັບລະບົບອື່ນໆເພື່ອປະໂຫຍດໃນການເຮັດວຽກໝາຍດ້ານ ຫຼື ມີການນຳໄປເຊື່ອມຕໍ່ເຂົ້າກັບເຕັກໂນໂລຊີອື່ນເພື່ອເພີ່ມຄວາມສາມາດໃຫ້ທຸກເຖິງກັບມະນຸດ, ດັ່ງຕົວຢ່າງ

1. Synface ການຊ່ວຍເຫຼືອການສົນທະນາທາງໂທລະສັບດ້ວຍໃບໜ້າຈໍາລອງ

- ເປັນຊອບແວທີ່ສາມາດສ້າງໃບໜ້າຈໍາລອງທີ່ສໍາພັນກັບການສົນທະນາຂອງຜູ້ທີ່ຢູ່ປາຍສາຍໂທລະສັບ, ເພື່ອຊ່ວຍເຫຼືອຜູ້ມີບັນຫາທາງການໄດ້ຢືນໄດ້. ພາບໃບໜ້າຈໍາລອງເຊິ່ງໃຫ້ພາບຄ້າຍໃບໜ້າຈຶ່ງຂອງບຸກຄົນທີ່ກຳລັງສົນທະນາຢູ່ນຳ, ເຮັດໃຫ້ຜູ້ເຫັນສາມາດເຂົ້າໃຈບົດສົນທະນາຈາກການອ່ານຮົບສິບໄດ້ເປັນຢ່າງດີ
- ຂຶນເຟັດ ໄດ້ຮັບການທິດສອບທີ່ສະຖາບັນຄົນຫຼູ້ໝວກໃນປະເທດອັງກິດ UK's Royal National Institute for the Deaf (RNID) ພົບວ່າ 84 % ຂອງຜູ້ທີ່ໄດ້ຮັບການທິດສອບສາມາດເຂົ້າໃຈບົດສົນທະນາ ແລະ ສາມາດລົມກັນທາງໂທລະສັບໄດ້ຢ່າງປິກກະຕິ

2. BEAM

- ສ້າງໂດຍ ມາກ ທຶນເດັນ (Mark W. Tilden) ນັກວິທະຍາສາດ ປະຈໍາຫຼອງທີ່ດີລອງແຫ່ງຊາດ LosAlamos ລັດ ນິວແມ່ກຊີໂໄກ, ສະຫະລັດອາເມລີກາ
- ສ້າງມາຈາກວົງຈອນອີເລັກໂທນິກຂະໜາດນັ້ອຍ, ໃຊ້ອຸປະກອນໜ້ອຍຮັນຈຶ່ງມີຂະໜາດນັ້ອຍແລະຮູບແບບການເຮັດວຽກບໍ່ຊັບຊອນ. ມີການເຕື່ອນໄຫວຄ້າຍຄືພິດຕິກຳຂອງສິ່ງມີຊີວິດ ເຊັ່ນ: ມິດແລະ ແມ່ງໄມ້ຕ່າງໆ
- " ບິມ " ໃຊ້ລະບົບຄວບຄຸມອີເລັກໂທນິກແບບງ່າຍ ແລະ " ທີ່ເຮັ້ນວ່າ " ເຄືອຂ່າຍເສັ້ນປະສາດ (Nervous Network) " ແກ່ນໄມ້ໂຄຣໂປຣເຊດເຊີ ເຊິ່ງເປັນຊຸດທານຊີດເຕີຫຼາຍໆໂຕທີ່ສາມາດຮັບ-ສື່ງຂໍ້ມູນຈາກໂຄງສ້າງຕົວຫຸ່ນ ແລະ ການເຕື່ອນໄຫວ. ຖ້າຂາຂ້າງໃດກະທິບມຳເຕີໄຟຟ້າ; ຈະເກີດແຮງໜ່ວງ ແລະ ປັບປຸງນວົງຈອນໂຄງໄຟຟ້າ, ເຮັດໃຫ້ຂາຂ້າງນັ້ນກ້າວໄປທາງອື່ນທັນທີ

3. ການຈຶ່ງຈໍາຕົວເລັກ 0-9 ໂດຍໃຊ້ ນິວອອນເມັດເວີກ

- ເປັນບົດນິພົນທີ່ສະເໜີການໃຊ້ຄວາມຄິດໃນການອອກແບບ ແລະ ສ້າງລະບົບຄອມພິວເຕີໃຫ້ມີໂຄງສ້າງທາງສະຖາປັດຕະຍະກຳຮູນແບບການເຮັດວຽກຂອງເຊວໃນສະໝອງມະນຸດ (Nerve cell) ຫຼື ນິວອອນ (Neural)
- ຊົງໂປຣແກຣມເພື່ອຮັບ input pattern ຂອງຕົວເລັກໃນຮູບບິດແມ່ນ ເປັນພາບຂາວດຳຂະໜາດ pattern 16 ຈຸດ 16 ພິກເຊລ ໃຊ້ຕົວເລັກຕົວພິມໃຫຍ່ ຕົວພິມນັ້ອຍ ຕົວພິມອງງ້າຂ້າຍ ແລະ ຕົວພິມອງງ້າຂວາຫັ້ງພິດ 33 pattern ໃຊ້ເປັນຖານຂໍ້ມູນໃນການສອນ ຈາກນັ້ນກໍ່ການກຳນົດນັ້ນກ້າວ, ຄ່າໄບແອດ, Layer, ພັງຊັນຕ່າງໆ ແລະ output ທີ່ເໝາະສົມ ແລ້ວຈະໄດ້ຄ່າອອກາມຄ່າຫຶ່ງ ຈາກນັ້ນນຳ input ທີ່ຕ້ອງການກວດສອບມາ Simulate ປຸງທຸງກັນວ່າມີຄ່າໄກວັງກັບຄ່າໄດ້ ເມື່ອ Simulate ແລ້ວໄດ້ຄ່າໄດ້ອອກ ມາສ້າງການໂຫຼດ pattern ຄໍາຕອບນັ້ນອອກາມສະແດງ

ບົດທີ 7 ລະບົບຜູ້ຂ່າງວຊານ

ລະບົບຜູ້ຂ່າງວຊານຈະຖືກໃຊ້ຮັດວຽກທີ່ຂັບຂອນໜ້າຍທີ່ສຸດ, ເຊິ່ງໃນອະດີດວຽກປະເພດນີ້ຈະສາມາດຮັດໄດ້ກໍ່ຕ້ອງອາໄສຜູ້ຂ່າງວຊານທີ່ເປັນມະນຸດເຫັນມັນ, ດ້ວຍວິທີການປະຍຸກໃຊ້ດ້ານບັນຍາປະດິດ. ລະບົບຜູ້ຂ່າງວຊານຈະຮັບເອົາຄວາມຮູ້ພື້ນຖານເຊິ່ງມະນຸດເປັນຜູ້ໃສ່ໃຫ້ ມາຮັດການປະມວນຜົນເຊັ່ນດູວກັບທີ່ມະນຸດແກ້ບັນຫາທີ່ຂັບຂອນ. ສິ່ງທີ່ດີທີ່ສຸດ ແລະ ມີປະສິດທິພາບໝາຍທີ່ສຸດຂອງລະບົບຜູ້ຂ່າງວຊານແມ່ນການບໍ່ມະຕິຄວາມຮູ້ນັ້ນເຮັດໄດ້ດີກວ່າລະບົບຊອບແວຄອມພິວເຕີກໍາມະດາ, ທີ່ໂດຍປົກກະຕິແລ້ວຈະອາໄສມະນຸດເປັນຜູ້ຕັດສິນໃຈ.

ລະບົບຜູ້ຊ່າງວຊານໄດ້ຖືກນຳໃຊ້ຢ່າງກວາງຂວາງໃນການວິໄຈ, ການວາງແຜນ, ການອອກແບບ, ການແປ, ການຄວບຄຸມ, ການບອກສະຖານະ, ການຄາດການ ແລະ ການອອກຄໍາສັ່ງໃນອະນາຄີດ. ດ້ວຍສະຖາບັດຕະຍະກຳສະໄໝໃໝ່ໃໝ່ຂອງຮາດແວ (Hardware) ທີ່ຖືກພັດທະນາໃຫ້ໃຊ້ໄດ້ໂດຍກົງກັບ ລະບົບຜູ້ຊ່າງວຊານ ແລະ ເຕັກໂນໂລຢີຂອງປັນຍາປະດິດລວມເຂົ້າກັນ. ຄວາມເປັນໄປໄດ້ທີ່ເຮັດໃຫ້ການ ພັດທະນາລະບົບສາມາດຮັດວຽກໄດ້ຖືກນມະນຸດຈຶ່ງມີຫຼາຍຍິ່ງຂຶ້ນ.

ການພັດທະນາລະບົບດັ່ງກ່າວຈະສາມາດເຮັດໃຫ້ເຄີຍບໍ່ພຽງແຕ່ມີລະບົບທີ່ມີຄວາມສາມາດຂຶ້ນເຖິງນັ້ນ ແຕ່ຢ່າງຈະເຮັດໃຫ້ເຄື່ອງສາມາດເຂົ້າໃຈເຄື່ອງທີ່ເກີດຂຶ້ນໄດ້ພ້ອມ.

7.1. ປະຫວັດຂອງລະບົບຜູ້ຊົ່ງວຊານ

ປະຫວັດຂອງລະບົບຜູ້ຊົ່ງວຊານແມ່ນເລີ່ມໃນຊ່ວງປີ 1969 ເຊັ່ນເວີດ ໄຟເຈັນບາມ (Edward Feigenbaum) ໄດ້ຮ່ວມກັບ ໂຈຊົ້ວ ລີເດີເບີກ (Joshua Lederberg) ຜູ້ທີ່ໄດ້ຮັບລາງວັນໂນບ່ອ (Nobel) ສາຂາຊີວະເຄມີ ແລະ ບຣຸດ ບຸຊານັນ (Bruce Buchanan) ໃນການພັດທະນາຊອບແວທີ່ໃຊ້ສໍາລັບການ ວິເຄາະໂຄງສ້າງໂມເລກຸນຂອງສານປະກອບທາງເຄມີໃນປີ ພສ 2508 ທີ່ມະຫາວິທະຍາໄລສະແຕນຝອດ ຊື່ DENDRAL ແລະ ລະບົບຊອບແວນີ້ການເປັນລະບົບຜູ້ຊົ່ງວຊານລະບົບທຳອິດຂອງໂລກ ທີ່ສາມາດ ເຮັດການວິເຄາະເພື່ອຄາດການໂຄງສ້າງໂມເລກຸນຂອງສານປະກອບ ໂດຍໃຊ້ຄໍາສ້າງ if-then ພື້ນຖານ ຈຳນວນໜຶ່ງທີ່ບອກເຖິງຄວາມແຕກຕ່າງຂອງອາຕອມໃນສານປະກອບ (Lindsay, R.K. et al., 1993)

ກາງປີ 1970 ເອດເວີດ ຊໍຕລິຟ (Edward Shortliffe) ນັກຝຶກ ແລະ ຄອມພິວເຕີໄດ້ພັດທະນາລະບົບຊອບແວຊະນິດໃໝ່ຂຶ້ນ MYCIN ທີ່ວິທະຍາໄລການແພດຂອງມະຫາວິທະຍາໄລສະແຕນຝອດ (Shortliffe, E.H., 1976) ເພື່ອໃຊ້ສໍາລັບການບົ້າມະຕິການຕິດເຊື້ອທີ່ສະໜອງຕໍ່ມາລະບົບນີ້ໄດ້ພັດທະນາກາຍເປັນເປື້ອກລະບົບຜູ້ຊ່ວຊານ (Expert System Shell) ແລະ ພັດທະນາເປັນລະບົບຜູ້ຊ່ວຊານລະບົບໃໝ່ ທີ່ໃຊ້ເຮັດການບົ້າມະຕິຄວາມຜິດປົກກະຕິຂອງປອດຊື່ວ່າ PUFF (Aikins, J.S., et al., 1982)

ໃນປີ ພ.ສ. 2524 ເອັດເວີດ ໄພເຈນບາມ ແລະ ໂລເບີດ ເອນເຈລ໌ມ (Robert Engelmore) ແລະ
ໝູ້ໄດ້ຮັ່ງກຳນົດຕັ້ງ ບໍລິສັດເຕັກໂນເລດ (Teknowledge) ຂຶ້ນ ຊຶ້ງເປັນບໍລິສັດທໍາອິດທີ່ເຮັດລະບົບຜູ້
ຊົ່ວຊານເພື່ອການຄ້າ

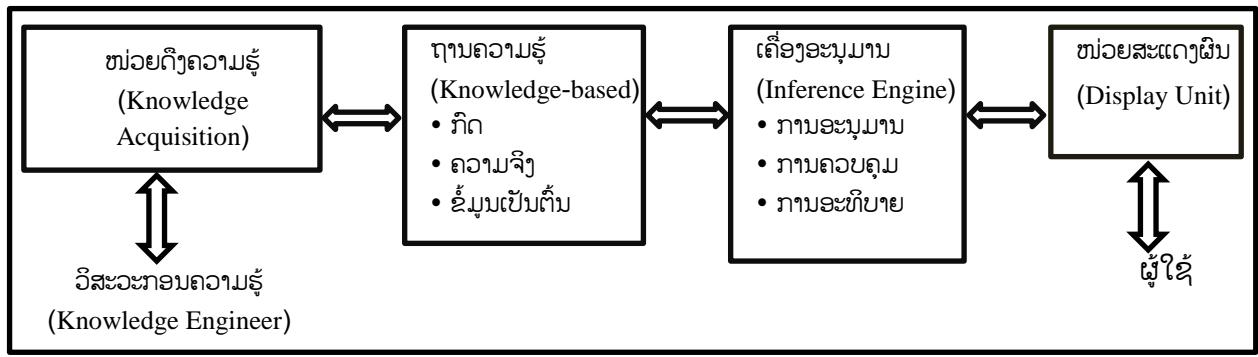
7.2. និយាមខទ្ធភាពជាអង់គ្លេស

ລະບົບຜູ້ຂ່າວຊານແມ່ນການເຮັດໃຫ້ຄອມພິວເຕີມີຄວາມສາມາດໃນການແກ້ປັນຫາທີ່ຊັບຊອນ ໄດ້ເຊັ່ນດູງວັນຍຸດທີ່ເປັນຜູ້ຂ່າວຊານ ຈະເຮັດແນວນັ້ນໄດ້ລະບົບຄອມພິວເຕີຈະຕ້ອງຈໍາລອງຂະບວນການຫາເຫດຜົນຂອງມະນຸດໂດຍອາໄສຄວາມຮູ້ ແລະ ການວິເຄາະ. ຕົວຢ່າງ: ການໃຊ້ລະບົບຜູ້ຂ່າວຊານສາມາດປຽບທຽບໄດ້ຕີກັບການໄປພົບທ່ານໝຶກອນເຮົາບໍ່ສະບາຍທ່ານໝຶກຈະຕັ້ງຄໍາຖາມແລ້ວໃຫ້ຄົນເຈັບຕອບ ແລະ ອາດຈະມີການກວດຮ່າງກາຍພ້ອມ ຈາກນັ້ນທ່ານໝຶກຈະບົ່ງມະຕີວ່າຄົນເຈັບເປັນພະຍາດຫຍັງ. ທ່ານໝຶກເຮັດແບບນັ້ນເພົ່າວ່າເພື່ອມີຄວາມຮູ້ດູງວັນຍຸດພະຍາດ, ເຊິ່ງການສອບຖາມອາການຂອງຄົນເຈັບແມ່ນເພື່ອໃຫ້ເປັນຂໍ້ມູນເພື່ອບົ່ງມະຕີພະຍາດດ້ວຍຄວາມຮູ້ທີ່ຕົນມີ. ໃນກໍລະນີນີ້, ຖ້າທ່ານໝຶກມີຄວາມຮູ້ຫຼາຍການບົ່ງມະຕີພະຍາດຈະມີຄວາມແນ່ນອນກວ່າທ່ານໝຶກມີຄວາມຮູ້ໜ້ອຍ. ໃນລະບົບຜູ້ຂ່າວຊານກໍເຊັ່ນດູງກັນ, ລະບົບຈະຖາມຄໍາຖາມຜູ້ໃຊ້ ແລະ ຜູ້ໃຊ້ຈະຕ້ອງຕອບຄໍາຖາມ ເມື່ອໝົດຄໍາຖາມແລ້ວ ຜົນການວິເຄາະຈະອອກມາເປັນຄໍາຕອບລາຍງານໃຫ້ຜູ້ໃຊ້ຮັບຮູ້. ໃນລະບົບຜູ້ຂ່າວຊານຖ້າລະບົບມີເງື່ອນໄຂ (rule) ແລະ ຄວາມຮູ້ຫຼາຍ ການບົ່ງມະຕີພະຍາດຈະມີຄວາມຖືກຕ້ອງກວ່າລະບົບມີຄວາມຮູ້ໜ້ອຍ. ລັກສະນະພາຍໃນຂອງລະບົບຜູ້ຂ່າວຊານຈະປະກອບດ້ວຍຄວາມສາມາດທີ່ສຳຄັນຕ່າງໆດັ່ງຕໍ່ໄປນີ້:

- ความรู้สังเขปที่มี domain ที่เริ่มต้นใจ
 - ภาระประมวลภาษาอังกฤษที่ถูกต้องมาก
 - ภาระประมวลภาษาอังกฤษที่ใช้วิธีเดา (heuristic) มาช่วยสังเคราะห์ข้อมูล
 - ความสามารถในการประมวลผลโดยใช้เครื่องมือเชิงคณิตศาสตร์ เช่น ลิสต์, แมตทริกซ์, ฟังก์ชัน, แคลคูลัส, สถิติ, และการวิเคราะห์ข้อมูล
 - ภาระประมวลผลโดยใช้เครื่องมือเชิงคณิตศาสตร์ เช่น ลิสต์, แมตทริกซ์, ฟังก์ชัน, แคลคูลัส, สถิติ, และการวิเคราะห์ข้อมูล
 - ภาระประมวลผลโดยใช้เครื่องมือเชิงคณิตศาสตร์ เช่น ลิสต์, แมตทริกซ์, ฟังก์ชัน, แคลคูลัส, สถิติ, และการวิเคราะห์ข้อมูล

7.3 ອົງປະກອບຂອງລະບົບຜູ້ຊ່າງວຊານ

ຈາກແນວຄວາມຄືດ ແລະ ການພະຍາຍາມທີ່ຈະອອກແບບການຄືດ, ການຈື້, ການປະມວນເຜີຍຂອງສະໜອງມະນຸດ ຈຶ່ງໄດ້ມີການອອກແບບລະບົບຜູ້ຊື່ວ່າງານທີ່ແບ່ງອອກເປັນສ່ວນໆ ດັ່ງສະແດງໃນຮູບທີ 7.1. ເຊິ່ງເປັນການສະແດງອົງປະກອບຂອງລະບົບຜູ້ຊື່ວ່າງານ ແລະ ເສັ້ນທີ່ເຊື່ອມຫາກັນດ້ວຍລູກສອນສະແດງເຖິງໜ່ວຍທີ່ຕົດຕໍ່ກັນຈາກສ່ວນຕ່າງໆ.



ຮູບທີ 7.1 ແຜນວາດຂອງລະບົບຜູ້ຊັ້ງວຊານ

7.3.1 ຖານຄວາມຮູ້ (Knowledge-Based)

ຖານຄວາມຮູ້ແມ່ນສ່ວນຂອງຄວາມຮູ້ທີ່ປະກອບດ້ວຍຄວາມຈິງ ແລະ ກົດຕ່າງໆທີ່ຖືກຈັດໄວ້ໃນລັກສະນະຂອງ heuristic ແລະ ມີລັກສະນະໃນການແກ້ບັນຫາສະເພາະບັນຫາໃດໜຶ່ງເຊັ່ນ: ຜູ້ຊັ່ງວຊານທີ່ກ່ຽວກັບການຮັກສາໂລກທົ່ວໃຈ, ໃນຖານຄວາມຮູ້ຈະປະກອບດ້ວຍກົດ ແລະ ຄວາມຈິງທີ່ກ່ຽວກັບເລື່ອງຂອງການຮັກສາໂລກທົ່ວໃຈ ເຊິ່ງກົດ ແລະ ຄວາມຈິງເງື່ອນນີ້ຈະຖືກຈັດວາງໄວ້ໃນຖານຄວາມຮູ້ໂດຍປົກກະຕິແລ້ວລະບົບຜູ້ຊັ່ງວຊານທີ່ດີຈະສ້າງຖານຄວາມຮູ້ແຍກອອກຈາກລະບົບ, ເພື່ອໃຫ້ຜູ້ສ້າງລະບົບຜູ້ຊັ່ງວຊານຈະໄສ່ຄວາມຮູ້, ເພີ່ມເຕີມ, ແກ້ໄຂ ຫຼື ປຸ່ຽນແປງຄວາມຮູ້ອື່ນພາຍຫຼັງ.

7.3.2 ເຄື່ອງອະນຸມານ (Inference engine)

ເຄື່ອງອະນຸມານແມ່ນສ່ວນທີ່ເຮັດໜ້າທີ່ໃນການປົງບໜູບຄວາມຮັກຕ່າງໆທີ່ຢູ່ໃນຖານຄວາມຮັກ ເພື່ອເຮັດໜ້າທີ່ໃນການຫາຜົນໄດ້ຮັບທີ່ເປັນໄປໄດ້. ໃນລະບົບຜູ້ຂ່າງວຊານເຄື່ອງອະນຸມານຈະເຮັດໜ້າທີ່ 2 ຢ່າງ
ຄື: 1. ເຮັດໜ້າທີ່ໃນການກວດສອບຄວາມຈິງ ແລະ ກົດທີ່ມີຢູ່ແລ້ວ ແລະ ເພີ່ມຄວາມຈິງອັນໃໝ່ເຂົ້າໄປ
ເມື່ອຈຳເປັນ ແລະ 2. ສ້າງການຕັດສິນໃຈກ່ຽວກັບລຳດັບກ່ອນ-ຫຼັງຂອງການອະນຸມານ. ໃນການປະຕິບັດ
ສອງໜ້າທີ່ມີເຄື່ອງຈະຕ້ອງເຮັດການຕິດຕໍ່ ແລະ ຂໍລັດປຶກສາກັບຜູ້ໃຊ້. ອົງປະກອບຂອງເຄື່ອງອະນຸມານ
ນັ້ນປະກອບດ້ວຍ 2 ສ່ວນຫຼາຍຄື: ສ່ວນທີ່ກ່ຽວກັບການອະນຸມານ (inference) ໃນການຫາຄວາມຮັກໃໝ່

ຈາກຄວາມຈິງ ແລະ ກົດທີ່ມີຢູ່ ແລະ ສ່ວນທີ່ກ່ຽວກັບການຄວບຄຸມ (control) ຈະເຮັດໜ້າທີ່ໃນການຄວບຄຸມ ແລະ ຈັດລຳດັບການອະນຸມານ

❖ ການອະນຸມານ

ໃນການອະນຸມານ, ເຕືອງອະນຸມານຈະອາໄສຫຼັກການຕ່າງໆດັ່ງຕໍ່ໄປນີ້ ເພື່ອເຮັດການອະນຸມານ:

1. **ໂມດດີໂພເນັ້ນ (Modus Ponens)** ແມ່ນຍຸດທະສາດໃນການອະນຸມານ ຫຼັກການຂອງໂມດດີໂພເນັ້ນມີວິທີການງ່າຍໆ ຄື: ຖ້າຫາກຮູ້ວ່າ A ຕີກ ແລະ ເມື່ອມີກົດທີ່ວ່າ “If A then B” ເຖິງຈະສາມາດສະຫຼຸບໄດ້ວ່າ B ຕີກ ຫຼື ອີກຄວາມໝາຍໝຶ່ງວ່າ “ເມື່ອພົບວ່າຟຣີມິດ (Premises) ຂອງກົດຖືກຕ້ອງກໍສາມາດເຊື່ອໄດ້ວ່າຂໍ້ສະຫຼຸບ (conclusion) ຂອງກົດຂຶ້ນນີ້ຖືກຕ້ອງດ້ວຍ”.

2. **ການຫາເຫດຜົນພາຍໃຕ້ຄ່າຄວາມເຊື້ອໝັ້ນ (Certainty factor)** ໃນກໍລະນີທີ່ມີຄວາມຮູ້ທີ່ບໍ່ສາມາດຕັດສິນໃຈວ່າຖືກຕ້ອງຮອຍເປີເຊັນ ເຕືອງອະນຸມານຈະເຮັດການອະນຸມານຄວາມຮູ້ນີ້ພາຍໃຕ້ຄວາມບໍ່ໝັ້ນຄວາມຮູ້ທີ່ບໍ່ໝັ້ນໃຈຈະຖືກກຳນົດໄວ້ດ້ວຍຄ່າຄວາມເຊື້ອໝັ້ນ (Certainty Factor) ທີ່ຂຽນແທນວ່າ cf, ເຊັ່ນ: ສີຂອງຫ້ອງຟ້າ = ສີຟ້າ cf 95 ໝາຍຄວາມວ່າເຮົາໝັ້ນໃຈວ່າ ສີຂອງຫ້ອງຟ້າເທົ່າກັບສີຟ້າ 95% (ຈາກ 100) ໃນການຫາເຫດຜົນພາຍໃຕ້ຄວາມໝັ້ນໃຈ ເນື່ອງຈາກວ່າໃນການອະນຸມານກົດຂໍຕ່າງໆຈະຕ້ອງມີຄວາມສຳພັນກ່ຽວເນື່ອງກັນ ດັ່ງນັ້ນເມື່ອມີການກຳນົດຄ່າຄວາມໝັ້ນໃຈໃຫ້ກັບ “ຄວາມຈິງ” ແລະ/ຫຼື “ກົດ” ອັນໄດ້ອັນໜຶ່ງຜົນຂອງມັນຈະໄປກ່ຽວຂ້ອງກັບ “ກົດ” ແລະ “ຄວາມຈິງ” ອັນອື່ນໆດ້ວຍ, ເຊັ່ນ:

Fact: ຫ້ອງຟ້າມີສີຟ້າ cf 80 (ໝັ້ນໃຈວ່າຫ້ອງຟ້າມີສີຟ້າ 80%)

Rule: if ຫ້ອງຟ້າມີສີຟ້າ then ອາກາດແຈ່ນໃສ

ຈາກ “ຄວາມຈິງ” ແລະ “ກົດ” ດັ່ງກ່າວເຮົາບໍ່ສາມາດໝັ້ນໃຈໄດ້ 100% ວ່າມີນີ້ “ອາກາດແຈ່ນໃສ” ເນື່ອງຈາກວ່າເຮົາໝັ້ນໃຈວ່າມີນີ້ “ຫ້ອງຟ້າມີສີຟ້າ” ດ້ວຍຄວາມໝັ້ນໃຈແຕ່ 80% ເທົ່ານັ້ນ (ຈາກ Fact) ຂະບວນການຫາເຫດຜົນພາຍໃຕ້ຄວາມບໍ່ແບ່ງໃຈ (Shortliffe, E.H. and Buchann, B.G., 1975) ນັ້ນຖືກພັດທະນາຂຶ້ນມາຄັ້ງທໍາອິດ ແລະ ໃຊ້ກັບລະບົບຜູ້ຊ່າງວຊານທີ່ຂື່ວ່າ MYCIN (Shortliffe, E.H., 1976) ມີຢູ່ໝາຍວິທີດັ່ງນີ້:

ການໄຮມກັນຂອງຄ່າຄວາມໝັ້ນໃຈ:

$$cf_{total} = cf_1 + \frac{cf_2(100 - cf_1)}{100}$$

ຕົວຢ່າງ:

1. if main-component = fish then best-color = white cf 50.
2. if sauce = tomato then best-color = white cf 70.
3. main-component = fish.
4. sauce = tomato.

ຜົນໄດ້ຮັບຂອງການຄໍານວນຄ່າຄວາມເຊື້ອໝັ້ນເທົ່າກັບ: best-color = white cf 85

ການກຳນົດສັດສວນຂອງຄວາມໝັ້ນໃຈ

$$cf_{total} = \frac{cf_1 * cf_2}{100}$$

ດັ່ງຕົວຢ່າງ:

1. main-component = meat cf 40.
2. if main-component = meat then best-color = red cf 50.

ຜົນໄດ້ຮັບຂອງການຄໍານວນຄ່າຄວາມໝັ້ນໃຈເທົ່າກັບ: best-color = red cf 20

❖ ການສ້າງອໍານາດຈຳແນກ (Resolution)

ອໍານາດຈຳແນກເປັນການພິສູດວ່າຄວາມຈີງທີ່ເກີດຂຶ້ນໃໝ່ນັ້ນເປັນຈີງ ຫຼື ບໍ່ ຈາກກຸ່ມຂອງຕັກກະສາດທີ່ມີຢູ່ແລ້ວ. ວິທີການຂອງອໍານາດຈຳແນກຈະອາໄສຫຼັກການຂອງຕັກກະສາດ

1. ໃນກໍລະນີທີ່ມີກົດດັ່ງນີ້ if A then B ຈະສາມາດປັບປຸງເປັນຕັກກະສາດໄດ້ເທົ່າກັບ not(A) or B
2. ໃນກໍລະນີທີ່ A ແລະ B ເປັນອີງປະກອບຂອງຕັກກະສາດ ຈະສາມາດສ້າງຕາຕະລາງຄ່າຄວາມຈີງໄດ້ດັ່ງນີ້:

A	B	$\sim(A)$	If A then B	$\sim(A) \text{ or } B$
T	T	F	T	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

3. ເມື່ອຮູ້ວ່າ $\sim(A)$ or B ແລະ ຮູ້ວ່າ A or C ຈະສາມາດສະໜູບປະໂຫຍກທັງສອງໄດ້ເປັນ B or C ພຽງປະໂຫຍກຄູງວໄດ້. ການປະຕິບັດແບບນີ້ເອີ້ນວ່າ ການສ້າງອໍານາດຈຳແນກ ເຊິ່ງມີຂັ້ນຕອນດັ່ງຕົວຢ່າງຕໍ່ໄປນີ້:

ຖ້າມີຄວາມຮູ້ທີ່ຕ້ອງການພິສູດ (ຈາກ Harmon, P. and King, D., 1985) ຕີ:

- 1: if (distance > 5 miles) then (mean=drive)
- 2: if (mean = drive) then (advice = take a cab)
- 3: fact: distance > 5 miles.

ການພິສູດ:

ຂັ້ນຕອນທີ 1: ປັບປຸງ if ... then ... ຂອງຄວາມຮູ້ທີ່ກຳນົດໃຫ້ເປັນຕັກກະສາດດັ່ງຕໍ່ໄປນີ້:

- 1: not (distance > 5 miles) or (mean=drive)
- 2: not (mean = drive) or (advice = take a cab)
- 3: fact: distance > 5 miles.

ຂັ້ນຕອນທີ 2: ໄສ່ສົມມຸດຖານທີ່ເຮົາຕ້ອງການຈະທົດສອບ, ສົມມຸດວ່າເຮົາຕ້ອງການຈະທົດສອບວ່າ ຖ້າເຮົາບໍ່ແນະນຳໃຫ້ “take a cab” ຜົນອອກມາຈະເປັນແນວໃດ.

- 4: not (advice = take a cab)

ចំណាំ 3: រៀនការងារទិន្នន័យដោយប្រើប្រាស់បញ្ជី:

not (distance > 5 miles) or (mean=drive)	ទាក់ទងទិន្នន័យ 1
not (mean = drive) or (advice = take a cab)	ទាក់ទងទិន្នន័យ 2
not (distance > 5 miles) or (advice = take a cab) (distance > 5 miles)	ទាក់ទងទិន្នន័យ 3
(advice = take a cab)	ទាក់ទងទិន្នន័យ 4
Not (advice = take a cab)	
Null	

ចំណាំ 4: សម្រាប់វាតាមលិមមុទាននេះខ្លួន មិនមានការងារទិន្នន័យដោយប្រើប្រាស់តាមរាយការណ៍ ពេលវេលាដែលបានបង្ហាញឡើង។ តែងនេះ ជឿនដែលបានបង្ហាញឡើងនៅពេលវេលាដែលបានបង្ហាញឡើង។ នៅពេលនេះ ទាក់ទងទិន្នន័យដោយប្រើប្រាស់តាមរាយការណ៍ នឹងបានបង្ហាញឡើង។

ទាក់ទងទិន្នន័យដោយប្រើប្រាស់តាមរាយការណ៍ នឹងបានបង្ហាញឡើង។ ការងារទិន្នន័យដោយប្រើប្រាស់តាមរាយការណ៍ នឹងបានបង្ហាញឡើង។ ទាក់ទងទិន្នន័យដោយប្រើប្រាស់តាមរាយការណ៍ នឹងបានបង្ហាញឡើង។ ទាក់ទងទិន្នន័យដោយប្រើប្រាស់តាមរាយការណ៍ នឹងបានបង្ហាញឡើង។

❖ ការងារទិន្នន័យ

ឱ្យរួមការងារទិន្នន័យទាំងអស់នៃការងារទិន្នន័យ 2 យ៉ាងតិច:

1. ការងារទិន្នន័យទាំងអស់នៃការងារទិន្នន័យ នឹងបានបង្ហាញឡើង។

2. ការងារទិន្នន័យទាំងអស់នៃការងារទិន្នន័យ នឹងបានបង្ហាញឡើង។ រាយការណ៍ នឹងបានបង្ហាញឡើង។

ឱ្យរួមការងារទិន្នន័យទាំងអស់នៃការងារទិន្នន័យ 2 យ៉ាងតិច នឹងបានបង្ហាញឡើង។ ការងារទិន្នន័យទាំងអស់នៃការងារទិន្នន័យ នឹងបានបង្ហាញឡើង។ ការងារទិន្នន័យទាំងអស់នៃការងារទិន្នន័យ នឹងបានបង្ហាញឡើង។

ការងារទិន្នន័យទាំងអស់នៃការងារទិន្នន័យ នឹងបានបង្ហាញឡើង។ ការងារទិន្នន័យទាំងអស់នៃការងារទិន្នន័យ នឹងបានបង្ហាញឡើង។ ការងារទិន្នន័យទាំងអស់នៃការងារទិន្នន័យ នឹងបានបង្ហាញឡើង។

ពិវឌ្ឍន៍ 7.1 ទាក់ទងទិន្នន័យទាំងអស់នៃការងារទិន្នន័យ នឹងបានបង្ហាញឡើង។ ទាក់ទងទិន្នន័យទាំងអស់នៃការងារទិន្នន័យ នឹងបានបង្ហាញឡើង។

Rule-1: if food = meat then color = red

Rule-2: if food = poultry then color = white

Rule-3: if food = fish then color = white

Rule-4: if sauce = yes and taste = sweet then flavor = sweet

Rule-5: if flavor = dry then sweetness = dry

Rule-6: if flavor = medium then sweetness = medium

Rule-7: if flavor = sweet then sweetness = sweet

Rule-8: if color = red and sweetness = dry then wine = cabernet-sauvignon

Rule-9: if color = red and sweetness = medium then wine = gamay

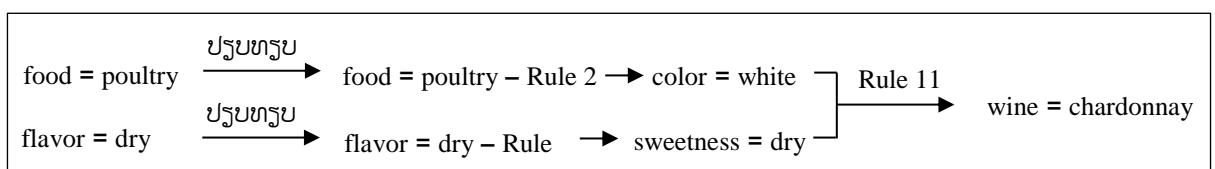
Rule-10: if color = red and sweetness = sweet then wine = burgundy

Rule-11: if color = white and sweetness = dry then wine = chardonnay

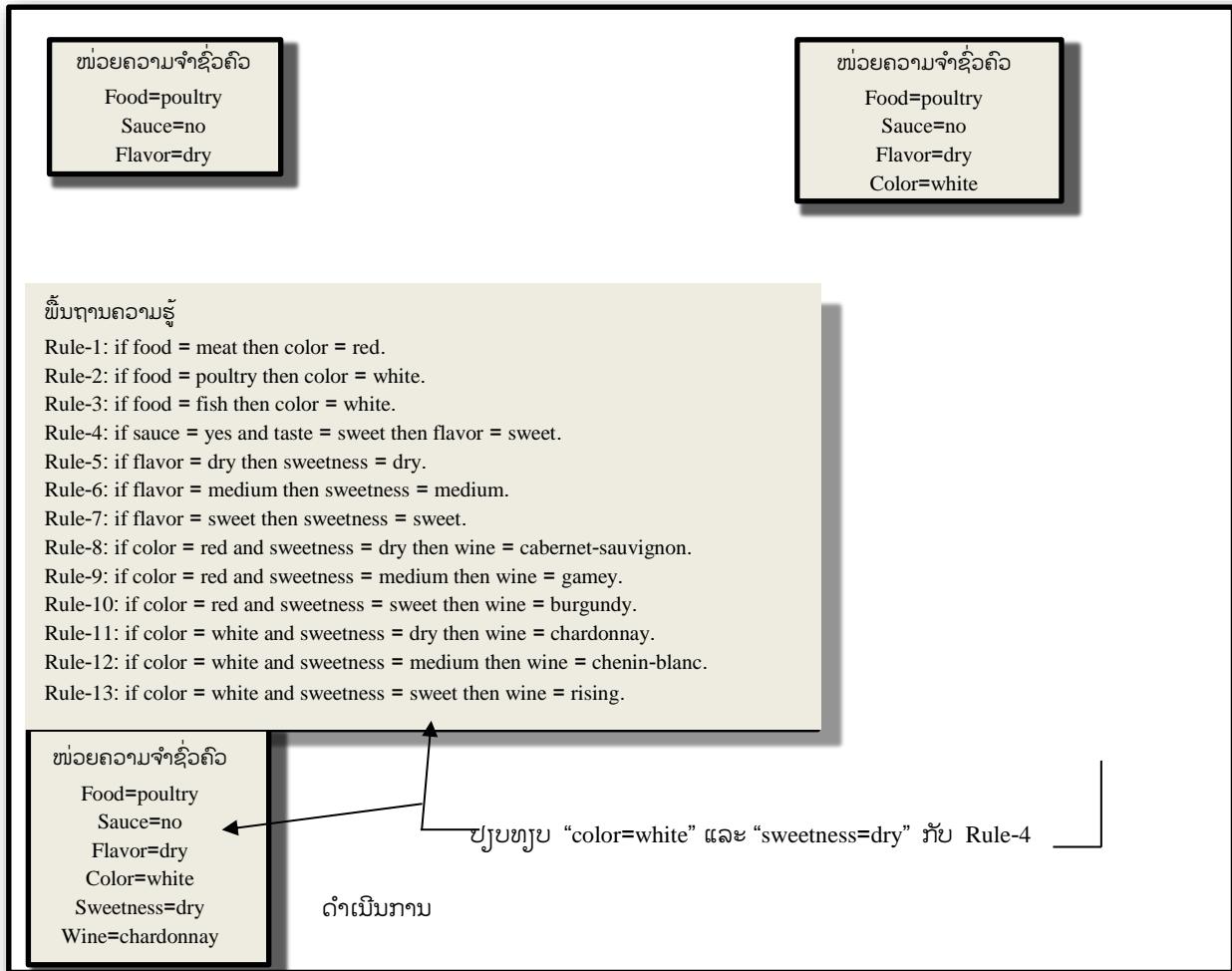
Rule-12: if color = white and sweetness = medium then wine = chenin-blanc

Rule-13: if color = white and sweetness = sweet then wine = Riesling

ການເຮັດວຽກຈະເລີ່ມຈາກສະຖານະປັດຈຸບັນທີ່ເກັບຢູ່ໃນໜ່ວຍຄວາມຈຳ ຕື: food = poultry
 e = no and flavor = dry ແລ້ວເອົາ food = poultry ໄປປູບຫງົບກັບກົດກ່ອນ ໄດຍເລີ່ມຈາກກົດ
 ອິດ. ການເລີ່ມຈາກກົດຂຶ້ນນີ້ເພົາວ່າຢູ່ສ່ວນຫຼັງ if ແມ່ນ food ເມືອປູບຫງົບແລ້ວເຮົາຈະເຫັນວ່າຂຶ້ນ
 ຜິດ ແລະ ທີ່ຖືກແມ່ນຂຶ້ທີ 2 (Rule-2) ເຮົາຈະໄດ້ສ່ວນທີ່ຢູ່ຫຼັງ then ແມ່ນ color = white ເປົ້ນ
 ຫຼູບໃໝ່ ຈາກນັ້ນເຮັດການກວດສອບ sauce = no ເຮັດໃຫ້ກົດຂຶ້ທີ 4 (Rule-2) ຜິດ ແລະ ບໍ່ມີກົດ
 ໃຫ້ກວດສອບອີກ. ດັ່ງນັ້ນໄປກວດສອບ flavor = dry ຈະໄດ້ກົດຂຶ້ທີ 5 (Rule-5) ຖືກ ເຮົາຈະໄດ້
 ນັກໃໝ່ທີ່ຢູ່ຫຼັງ then ຕື: sweetness = dry ໄປໃຊ້ຕໍ່. ຂໍສະຫຼຸບໃໝ່ທີ່ໄດ້ຕື: color = white ແລະ
 taste = dry ແມ່ນເປົ້າໝາຍຍ່ອຍທີ່ເຮົາຈະເອົາໄປປູບຫງົບຕໍ່. ເລີ່ມຈາກ color = white ເຫັນວ່າ
 ມີຄໍາວ່າ ແມ່ນກົດທີ 8 (Rule-8) ເຖິງ ກົດທີ 13 (Rule-13) ເຊິ່ງ ກົດທີ 11 (Rule-11) ແມ່ນຂຶ້ນ
 ທີ່ຖືກເພາະມີ color = white ຕໍ່ໄປປູບຫງົບ sweetness ຂອງກົດທີ 11 ຈະໄດ້ sweetness =
 ຖາມຂໍສະຫຼຸບ. ດັ່ງນັ້ນກົດຂຶ້ທີ 11 ຈຶ່ງຖືກ ເຮົາຈະໄດ້ wine = chardonnay ຈົບການເຮັດວຽກຂອງ
 ປິນການອະນຸມານໄດ້ສະແດງທີ່ຮູບທີ 7.2 ແລະ ຜັງງານຂອງການດຳເນີນການສະແດງທີ່ຮູບນີ້



ຮູບທີ 7.2 ສະຫຼຸບຜົນຂອງການອະນຸມານແບບ



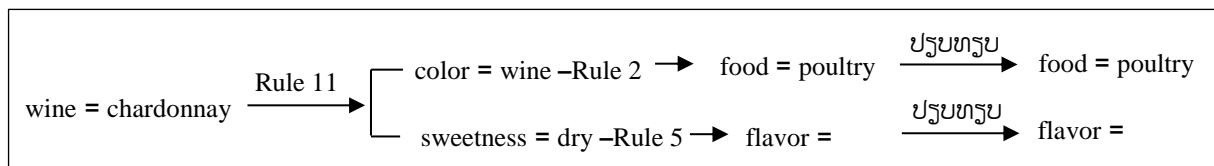
ຮັບທີ 7.3. ຜັງວິກຂອງການອະນຸມານແບບໄປທາງໜ້າ

ຕົວຢ່າງທີ 7. 2 ຈາກຖານຄວາມຮູ້ທີ່ເປັນກົດເບື້ອງຕົ້ນ ຖ້າກຳນົດວ່າ food = poultry, sauce = no and flavor = dry ໃຫ້ແນະນຳວ່າຈະເລືອກ wine ແບບໃດ ໂດຍໃຊ້ວິທີແບບກັບຫ້າ.

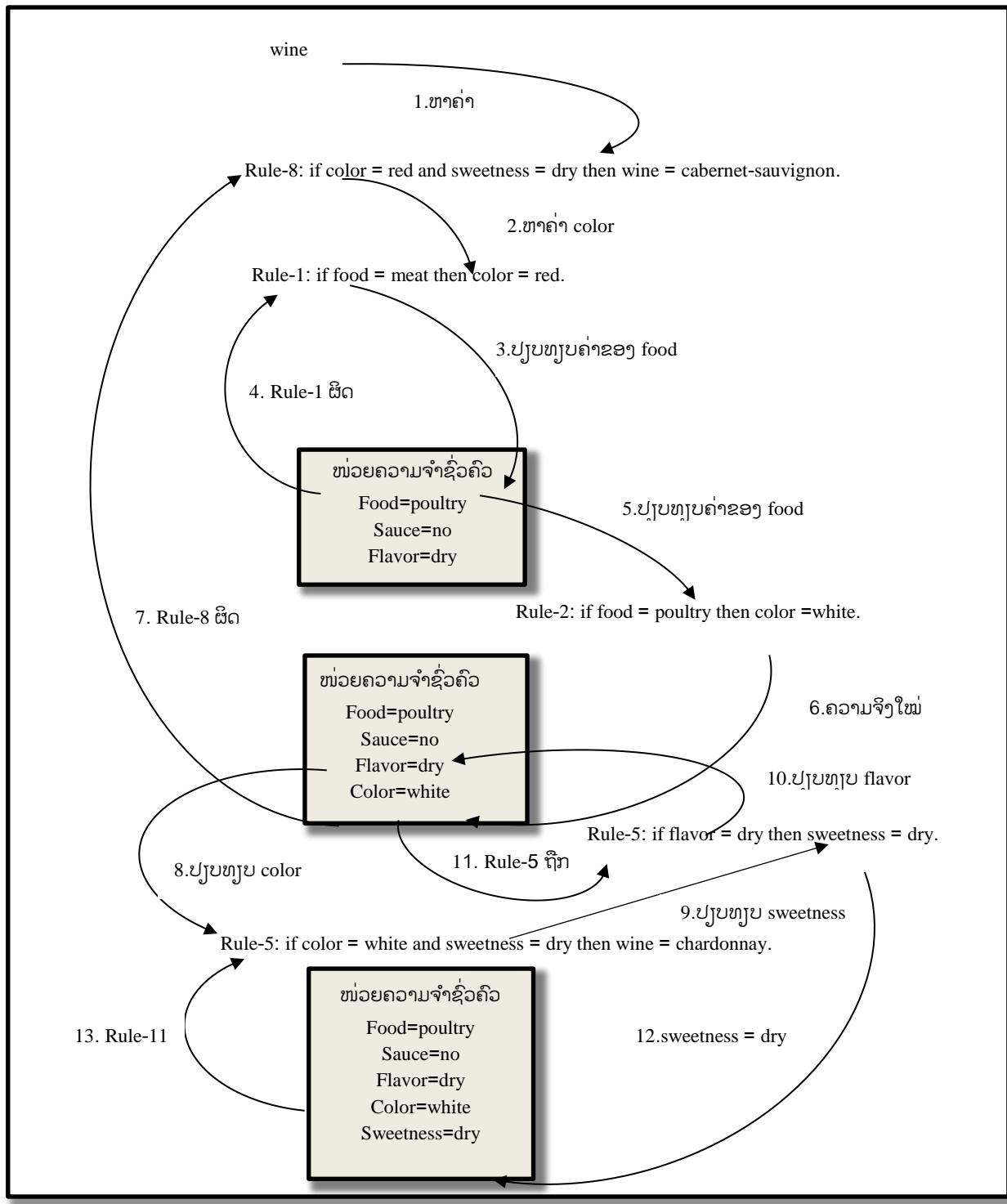
ການອະນຸມານຈະເລີ່ມຈາກການຄົ້ນຫາຄ່າຂອງ wine ເຊິ່ງມາຈາກ goal = wine ໃນຖານຄວາມຮູ້ ດ້ວຍການຫາວ່າມີກົດຂຶ້ນໄດ້ແດ່ທີ່ຫຼັງ then ມີການສະຫຼຸບກູ່ວັກບເລື່ອງ wine. ເນື້ອສຳຫຼວດກໍ່ເຫັນວ່າ ກົດຂຶ້ນທີ 8 (Rule-8) ທີ່ຂຽນວ່າ “Rule-8: if color = red and sweetness = dry then wine = cabernet-sauvignon” ຈາກນັ້ນເຮັດການສຳຫຼວດຫຼັງ if ແລ້ວເຮັດການພິສູດວ່າ color = red ເຮັດການພິສູດຫຼັງ then ຕໍ່ມາທີ່ກົດຂຶ້ນທີ 1 ດັ່ງນີ້: “Rule-1: if food = meat then color = red” ໃຫ້ຫາຄ່າຂອງ food ເຊິ່ງໄດ້ food = poultry ຈາກຄວາມຈິງທີ່ກໍານົດໃຫ້ໃນຕອນທຳອິດ ເຮັດໃຫ້ກົດຂຶ້ນທີ 1 ຜິດ ເຮັດໃຫ້ການເອີ້ນກົດຂຶ້ນທີ 2 “Rule-2: if food = meat then color = white”.

ເຊິ່ງຈະໄດ້ food = poultry ທີ່ຖືກຕ້ອງ, ດັ່ງນັ້ນກົດຂໍ້ທີ 2 ຖືກຕ້ອງ ແລະ ເຮົາຈະໄດ້ຂໍ້ສະຫຼຸບໃໝ່ທີ່ວ່າ color = white ເຊິ່ງເຮັດໃຫ້ກົດຂໍ້ທີ 8 ທີ່ກຳລັງພິສູດວ່າ color = red ຜິດທັນທີ. ຈາກນັ້ນເຮັນກົດຂໍ້ທີ 9 “Rule-9: if color = red and sweetness = medium then wine = gamay” ຫຼືມີ color = red ກໍຜິດອີກ. ເຮັນກົດຂໍ້ທີ 10 “Rule-10: if color = red and sweetness = sweet then wine =

burgundy” កំដើរិភាគ. ទាន់ន័យពីកិត្យទាំង 11 “Rule-10: if color = white and sweetness = dry then wine = chardonnay” ត្រូវមិន color = white ឡើងបានដឹងថាសំណើន៍ទាំង 11 ត្រូវតែង. ទាន់ន័យពីកិត្យទាំង 11 ត្រូវបានដឹងថាសំណើន៍ទាំង 4 “Rule-4: if sauce = yes and taste = sweet then sweetness = sweet” ត្រូវកិត្យទាំង 4 ដឹងថាសំណើន៍ទាំង 4 ត្រូវតែង. ទាន់ន័យពីកិត្យទាំង 4 ត្រូវបានដឹងថាសំណើន៍ទាំង 5 មានស្មើគ្នាដូចតុល្យ sweetness ព័ត៌មាន “Rule-5: if flavor = dry then sweetness = dry” ត្រូវកិត្យទាំង 5 ត្រូវតែង និងត្រូវតែង ទាន់ន័យពីកិត្យទាំង 5 ត្រូវតែង និងត្រូវតែង ទាន់ន័យពីកិត្យទាំង 7.4 និងត្រូវតែង ទាន់ន័យពីកិត្យទាំង 7.5.



នូបទី 7.4 សម្រួលដឹងទូទៅរបស់ការអនុវត្តន៍



ຮບທີ 7.5.ຜັງວູກຂອງການອະນຸມານແບບກັບຄືນ

ລວກສະນະສະເພາະອີກແບບໜຶ່ງຂອງເຄື່ອງອະນຸມານທີ່ຈະຕ້ອງພິຈາລະນາກໍ່ຕີ ເຄື່ອງອະນຸມານເປັນແບບໄມ້ໂນໂທນິກ (Monotonic) ຫຼືແບບນັ້ນໄມ້ໂນໂທນິກ (Nonmonotonic). ການຫາຄວາມຈິງແບບໄມ້ໂນໂທນິກຕີ ຄວາມຈິງໄດ້ທີ່ຖືກສໍາຫຼວດແລ້ວວ່າເປັນຈິງ ຄວາມຈິງນັ້ນຈະຄົງຢູ່ຕະຫຼອດໄປຈົນກວ່າການໃຫ້ຄໍາປຶກສາຂອງລະບົບຈະສັນສົດ. ຕົວຢ່າງຂອງຄວາມຮູ້ທີ່ຕ້ອງອະນຸມານແບບນີ້, ເຊັ່ນ: ຈາກຄວາມຈິງທີ່ວ່າ “ຄວາມໄວແສ່າຂອງພິມເປັນ 100” ຄ່າຂອງຄວາມໄວແສ່າທີ່ເກີ່າກັບ 100 ນີ້ຈະ

ເປັນຄວາມຈິງຕະຫຼອດການໃຫ້ຄໍາປຶກສາ. ສໍາລັບການທາເຫດຜົນແບບນັ້ນ ໂມໂນໄທນິກສີ ຄວາມຈິງໄດ້ທີ່ເລີຍຖືກກວດສອບແລ້ວວ່າເປັນຈິງ ຄວາມຈິງນີ້ຍັງສາມາດຈະປ່ຽນແປງຕໍ່ໄປໃນພາຍຫຼັງໄດ້. ຕົວຢ່າງ ຂອງຄວາມຮູ້ທີ່ຕ້ອງອະນຸມານແບບນີ້ຄື: ການສືບສວນສອບສວນຂອງຕໍ່ຫຼວດໃນການສືບສວນຊ່ວງທໍາອິດມີຄວາມເປັນໄປໄດ້ວ່າຂໍ້ມູນທີ່ມີຢູ່ຍັງບໍ່ຈະແຈ້ງ ຜູ້ຕ້ອງຫາອາດມີໄດ້ຫຼາຍຄົນ ແຕ່ຫຼັງຈາກທີ່ໄດ້ຮັບຂໍ້ມູນໃໝ່ງເຂົ້າມາໃນພາຍຫຼັງແລ້ວ ຜູ້ຕ້ອງຫາອາດຈະປ່ຽນແປງໄປກໍໄດ້.

❖ ຫ່ວຍດຶງຄວາມຮູ້ (Knowledge Acquisition Unit)

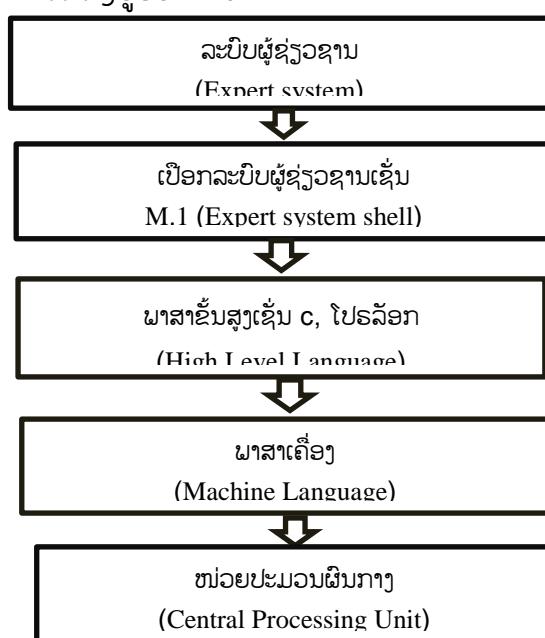
ໜ່ວຍດຶງຄວາມຮູ້ເປັນໜ່ວຍທີ່ຈະຮັບຄວາມຮູ້ຈາກຜູ້ຊ່ວຊານ ຫຼື ວິສາວະກອນຄວາມຮູ້ ເມື່ອວິສາວະກອນສະແດງຄວາມຮູ້ ຫ່ວຍດຶງຄວາມຮູ້ໃນລະບົບຜູ້ຊ່ວຊານຈະເຮັດໜ້າທີ່ແປກົດຄວາມຈິງ, ຂໍສົມມຸດຖານ ແລ້ວງປະກອບອື່ນໆຂອງຄວາມຮູ້ທີ່ມີຢູ່ແຍກຈາກລະບົບເຂົ້າສູ່ຖານຄວາມຮູ້ທີ່ສາມາດປະມວນຜົນໄດ້ ຂອງລະບົບ.

❖ ຫ່ວຍອະທິບາຍ (Explanation Unit)

ໜ່ວຍອະທິບາຍແມ່ນໜ່ວຍທີ່ຄອຍອະທິບາຍ ແລະ ໃຫ້ເຫດຜົນໃນການອະນຸມານ ໃນລະຫວ່າງທີ່ຜູ້ໃຊ້ເຄື່ອງກຳລັງສົນທະນາຢູ່ກັບລະບົບຜູ້ຊ່ວຊານນັ້ນ. ຜູ້ໃຊ້ສາມາດຖາມທາເຫດຜົນໄດ້ວ່າເປັນຫຍັງຈິງຕັ້ງຄໍາຖາມແບບນັ້ນ ແລະ ຜູ້ໃຊ້ແມ່ນຜູ້ຕ້ອງການຂໍຄໍາປຶກສາກັບລະບົບຜູ້ຊ່ວຊານ.

7.3.3 ພາສາ ແລະ ເຄື່ອງມື

ໃນລະບົບຜູ້ຊ່ວຊານມີແນວຄົດກ່ຽວກັບເລື່ອງພາສາ ແລະ ເຄື່ອງມືທີ່ແຕກຕ່າງຈາກແນວຄົດເກົ່າງຫຼາຍຢ່າງ, ພາສາໃນລະບົບຜູ້ຊ່ວຊານຈະໝາຍເຖິງພາສາຂັ້ນສູງ (High Level Language) ທີ່ໃຊ້ໃນການສ້າງລະບົບຜູ້ຊ່ວຊານ, ເຊັ່ນ: ລີສປ (LISP) ແລະ ໂປຣລ້ອກ (PROLOG) ແລະ ເຄື່ອງມື (Tools) ໝາຍເຖິງເຄື່ອງມືທີ່ໃຊ້ໃນການສະແດງຄວາມຮູ້ໃຫ້ກັບລະບົບຜູ້ຊ່ວຊານ, ເຊັ່ນ: EMYCIN ແລະ M.1. ໝາຍຄວາມວ່າພາສາເປັນສ່ວນທີ່ໃຊ້ໃນການສ້າງເຄື່ອງມື ຖ້າຈະສະແດງລະດັບຂອງຊອບແວອອກມາເປັນຂັ້ນໆ ຈະໄດ້ດັ່ງຮູບທີ່ 7.6



ຮູບທີ່ 7.6 ລະດັບຕ່າງໆຂອງຊອບແວໃນລະບົບຜູ້ຊ່ວຊານ

ຈາກຮູບທີ 7.6 ທີ່ສະແດງລະບົບຊອບແວທີ່ມີລະດັບຕ່າງໆ ທີ່ໃຊ້ໃນການສ້າງເປືອກ (shell) ລະບົບຜູ້ຂ່າຍວຊານ M.1 ຈະເຫັນໄດ້ວ່າພາສາຂັ້ນສູງທີ່ໃຊ້ໃນການສ້າງຄິພາສາ C ແລະ ຄວາມຮູ່ກໍ່ໃຊ້ເປືອກລະບົບຜູ້ຂ່າຍວຊານເປັນເຄື່ອງມືເຊັ່ນ: M1. ເປືອກລະບົບຜູ້ຂ່າຍວຊານໃນປັດຈຸບັນມີຫຼາຍຊະນິດ ເຊັ່ນ :Level 5 ເປັນເປືອກລະບົບຜູ້ຂ່າຍວຊານທີ່ໃຊ້ການສະແດງຄວາມຮູ່ໂດຍອາໄສສົເງມ (frame) ເປັນຖານ , M.1 ໃຊ້ກົດເປັນຖານ. ດັ່ງນັ້ນ, ໃນລະບົບຜູ້ຂ່າຍວຊານໜຶ່ງ ຈະໃຊ້ພາສາຂັ້ນສູງເຊັ່ນ: LISP ແລະ PROLOG, ເປັນໂຕສ້າງເຄື່ອງມືເຊັ່ນ: EMYCIN ແລະ M.1. ຈາກນັ້ນກໍ່ໃຊ້ຊື່ເຄື່ອງມືໃນການສະແດງຄວາມຮູ່ມີຊື່ເອັ້ນພິເສດ, ແຕ່ສໍາລັບຜູ້ທີ່ໃຊ້ເຄື່ອງມືໃນການສະແດງຄວາມຮູ່ມີຊື່ເອັ້ນພິເສດວ່າ “ວິສະວະກາອນຄວາມຮູ່”

ການສ້າງລະບົບຜູ້ຂ່າຍວຊານໃນຍຸກທຳອິດບໍ່ໄດ້ມີການແຍກເຄື່ອງມືອອກມາຈາກຕົວລະບົບ (Shortliffe, E.H., 1976) MYCIN ເປັນລະບົບຜູ້ຂ່າຍວຊານລະບົບທຳອິດທີ່ຖືກສ້າງຂຶ້ນມາໂດຍຄະນະຂອງນັກວິຊາການຈາກມະຫາວິທະຍາໄລສະແຕນຝອດ (Stanford) ເປັນໂປຣແກຣມທີ່ໃຫ້ຄໍາປຶກສາກ່ຽວກັບການຮັກສາໂລກທົ່ວໃຈກັບແພດທີ່ຖືກສ້າງຂຶ້ນມາພາຍໃຕ້ພາສາລືສປ ເນື່ອງຈາກ MYCIN ສາມາດຮັດວຽກໄດ້ສະເພາະກັບການຮັກສາໂລກທົ່ວໃຈເທົ່ານັ້ນ ເຊິ່ງເປັນຂໍ້ຈຳກັດໜ້າຍ MYCIN ຈຶ່ງຖືກພັດທະນາຕໍ່ມາອີກໂດຍການເອົາຄວາມຮູ່ກ່ຽວກັບໂລກທົ່ວໃຈອອກ ແລ້ວຮັດໃຫ້ຜູ້ໃຊ້ໃສ່ຄວາມຮູ່ຂໍ້ງກໍໄດ້ MYCIN ຈຶ່ງກາຍເປັນຊອບແວໂຕໃໝ່ທີ່ເອັ້ນວ່າ EMYCIN ຫຼື Empty MYCIN ຫຼື MYCIN ວ່າງເຢົ່າ. EMYCIN ຈຶ່ງກາຍເປັນເຄື່ອງມືຂອງລະບົບຜູ້ຂ່າຍວຊານຕົວທຳອິດ. ຫຼັງຈາກມີການພັດທະນາ EMYCIN ຕໍ່ມາເຄື່ອງມືທີ່ມີລັກສະນະເຊັ່ນດູວກັບ EMYCIN ໄດ້ເກີດຂຶ້ນຢ່າງໜ້າຍໜ້າ ຕົວຢ່າງເຄື່ອງມືໃນປັດຈຸບັນຄື: Level-5 Exsys ແລະ OPS.

ເນື້ອເວົ້າເຖິງພາສາ ແລະ ເຄື່ອງມືກັບລະດັບຕ່າງໆ ຂອງຊອບແວເຫຼົ້ານັ້ນແລ້ວ ບາງຄັ້ງການຈຳແນກລະດັບຂອງສິ່ງຕ່າງໆເຫຼົ້ານີ້ຈະມີຄວາມສັບສົນພໍສົມຄວນມີຊອບແວບາງຊະນິດທີ່ບໍ່ໄດ້ເປັນພາສາຂັ້ນສູງເຊັ່ນດູວກັບລືສປ ເຊິ່ງໃຊ້ວົງກົງງາງແຕ່ດ້ານໂປຣແກຣມເທົ່ານັ້ນ, ແຕ່ກໍບໍ່ໃຊ້ເຄື່ອງມືທີ່ຈະສາມາດສະແດງຄວາມຮູ່ໄດ້ຢ່າງດີເຊັ່ນດູວກັບ M.1 ຫຼື EMYCIN ເພື່ອຄວາມສະດວກຊອບແວເຫຼົ້ານີ້ຈຶ່ງຖືກຈັດເປັນສິ່ງທີ່ເອັ້ນວ່າ: ສິ່ງແວດລ້ອມ (Environment) ຊອບແວທີ່ມີລັກສະນະເປັນສິ່ງແວດລ້ອມໄດ້ແກ່ OPS5.

ຖ້າຈະພິຈາລະນາກັນໃຫ້ລະອຽດຂຶ້ນໄປອີກ ເນື້ອປູປທູບກັບລືສປກັບໂປຣລັອກແລ້ວ ໂປຣລັອກເປັນພາສາຂັ້ນສູງທີ່ສາມາດນຳມາສ້າງລະບົບຜູ້ຂ່າຍວຊານໄດ້ງ່າຍກວ່າລືສປ ຍ້ອນວ່າໂປຣລັອກຈະສາມາດສະແດງຄວາມຮູ່ແບບງ່າຍ່າໄດ້ ໃນຂະນະທີ່ລືສປເປັນພົງງາສາໂປຣແກຣມທຳມະດາ. ດັ່ງນັ້ນ, ໂປຣລັອກຈະມີລັກສະນະໄກສິ່ງແວດລ້ອມໜ້າຍກວ່າລືສປ.

ໃນການເລືອກຊອບແວເພື່ອສ້າງເຄື່ອງມືໃຫ້ກັບລະບົບຜູ້ຂ່າຍວຊານ ຍັງມີການຖູ່ງກັນໜ້າຍວ່າຈະເລືອກຊອບແວທີ່ເປັນລັກສະນະສິ່ງແວດລ້ອມ ຫຼືໄກສິ່ງແວດລ້ອມ, ເຊັ່ນ: ໂປຣລັອກ, OPS5 ຫຼື ຈະເລືອກພາສາຂັ້ນສູງຢ່າງເຊັ່ນ ລືສປ ຫຼື ຈະເລືອກຊອບແວທີ່ໄກເຄື່ອງເຊັ່ນ: KEE, LOOPS ແຕ່ມີສິ່ງທີ່ນໍາສັງເກັດໃນການພິຈາລະນາຢ່າງໜຶ່ງຄື ຊອບແວທີ່ເຂົ້າໄກພາສາຂັ້ນສູງຈະຮັດໃຫ້ສາມາດສ້າງເຄື່ອງມືທີ່ມີ

ຄວາມຄ່ອງຕົວສູງກວ່າຊອບແວທີ່ເຂົ້າໄກເຄື່ອງມື ແຕ່ຊອບແວທີ່ເຂົ້າໄກເຄື່ອງມືຈະສ້າງໄດ້ງ່າຍກວ່າຊອບແວທີ່ເຂົ້າໄກພາສາຂັ້ນສູງ.

ບົດທີ 8 ການພັດທະນາລະບົບຜູ້ຂໍ່ວຊານ

8.1. ການພັດທະນາລະບົບຜູ້ຂໍ່ວຊານ

ໃນການພັດທະນາລະບົບຜູ້ຂໍ່ວຊານຈະເປັນການສະແດງຄວາມຮູ້ໂດຍອາໄສເປື້ອກລະບົບຜູ້ຂໍ່ວຊານ (Expert System Shell) ເປັນວິທີການເກັບຄວາມຮູ້ໃຫ້ກັບລະບົບຜູ້ຂໍ່ວຊານເທົ່ານັ້ນ, ບໍ່ໄດ້ໝາຍເຖິງການສ້າງຊອບແວທີ່ໃຊ້ໃນການເກັບຄວາມຮູ້, ເຊິ່ງເປັນໜ້າທີ່ຂອງນັກຂຽນໂປຣແກຣມທີ່ຈະພັດທະນາຂຶ້ນມາ.

ອີກຢ່າງໜຶ່ງທີ່ຈະຕ້ອງເຂົ້າໃຈກໍສືລະບົບຜູ້ຂໍ່ວຊານທີ່ເຄີຍກ່າວເຖິງແມ່ນລະບົບຜູ້ຂໍ່ວຊານທີ່ສາມາດໃຫ້ຄໍາປຶກສາໄດ້ສະເພາະເລື່ອງ, ເຊິ່ງຈະເປັນການຍາກໝາຍທີ່ຈະພັດທະນາລະບົບຜູ້ຂໍ່ວຊານທີ່ໃຫ້ຄໍາປຶກສາໃນໝາຍໆເລື່ອງພາຍໃນທານຄວາມຮູ້ອັນດູວກັນ. ໃນການພັດທະນາລະບົບຜູ້ຂໍ່ວຊານທີ່ຈະກ່າວເຖິງຕໍ່ໄປນີ້ ຈະກ່າວເຖິງລັກສະນະການພັດທະນາໃນ 2 ລັກສະນະຕີ: ການພັດທະນາລະບົບທີ່ມີຂະໜາດນ້ອຍ ແລະ ຂະໜາດໃຫ່ຍ ເຊິ່ງມີລັກສະນະຂອງບັນຫາເປັນການບົ່ງມະຕິຫຼາຍກວ່າການສັງເຄາະ ແລະ ຕົວຢ່າງທີ່ຍົກມາເພື່ອປະກອບການອະທິບາຍຈະເນັ້ນໃສ່ການບົ່ງມະຕິບັນຫາເທົ່ານັ້ນ. ບັນຫາສ່ວນໃຫ່ຍຈະເປັນບັນຫາທີ່ສາມາດປະເມີນຜົນຂອງການປົງບໜງບໄດ້ ແລະ ສາມາດກຳນົດຄໍາຕອບທີ່ຄົບຖ້ວນໄດ້. ສໍາລັບວິທີການສັງເຄາະເຊິ່ງເປັນບັນຫາທີ່ມີລັກສະນະການຄາດໝາຍຄໍາຕອບບໍ່ໄດ້ ຈະບໍ່ໄດ້ກ່າວເຖິງໃນປົດນີ້. ລະບົບຜູ້ຂໍ່ວຊານຂະໜາດນ້ອຍໝາຍເຖິງລະບົບທີ່ມີກົດ (ໃນກໍລະນີທີ່ເປັນ Rule-based) ບໍ່ຫຼາຍທຸາຈະສະແດງຄວາມຮູ້ໃນແບບຮູບຂອງກົດ ຫຼື ຂະໜາດຄວາມຮູ້ທີ່ປະມານກັນເມື່ອສະແດງຄວາມຮູ້ໂດຍວິທີການອື່ນ. ການຕັ້ງກົດເການວ່າລະບົບຂະໜາດນ້ອຍຄວນມີກົດຈັກຂັ້ນຍັງບໍ່ທັນໄດ້ກຳນົດຕາຍຕົວ, ແຕ່ໄດ້ຫຼັກສ່ວນໃຫ່ຍແລ້ວຄວນຄໍານີ້ເຖິງຂະໜາດຂອງຄວາມຮູ້ ແລະ ຄວາມສັບຊ້ອນຂອງຄວາມຮູ້ພອມ. ກົດເການງ່າຍໆທີ່ພໍຈະເຂົ້າໃຈໄດ້ລະຫວ່າງລະບົບຜູ້ຂໍ່ວຊານຂະໜາດນ້ອຍ ແລະ ຂະໜາດໃຫ່ຍມີຄວາມແຕກຕ່າງກັນທີ່ເຫັນໄດ້ຈະແຈ້ງຕີ ຂອບເຂດຂອງຄວາມຮູ້ມີຄວາມກວາງໃຫ່ຍຂະໜາດໃດ.

ໂດຍຫຼັກສ່ວນແລ້ວການພັດທະນາລະບົບຜູ້ຂໍ່ວຊານຫຼຸກຂະໜາດມີຫຼັກການດູວກັນ ອາດຈະມີຄວາມແຕກຕ່າງກັນໃນທາງຂອງວົງຈອນການພັດທະນາ ແລະ ລາຍລະອຽດບາງຢ່າງເທົ່ານັ້ນ. ເນື່ອງຈາກວ່າການພັດທະນາລະບົບຜູ້ຂໍ່ວຊານຂະໜາດນ້ອຍ ລະດັບຄວາມສັບຊ້ອນຈະນ້ອຍກວ່າເມື່ອຫຼັງບໍກັບລະບົບທີ່ມີຂະໜາດໃຫ່ຍ. ດັ່ງນັ້ນ, ຂັ້ນຕອນບາງຢ່າງທີ່ຈະໃຊ້ໃນການພັດທະນາລະບົບຈົ່ງມີຄວາມຈຳເປັນເພາະຈະເປັນການສິ້ນເປື້ອງ.

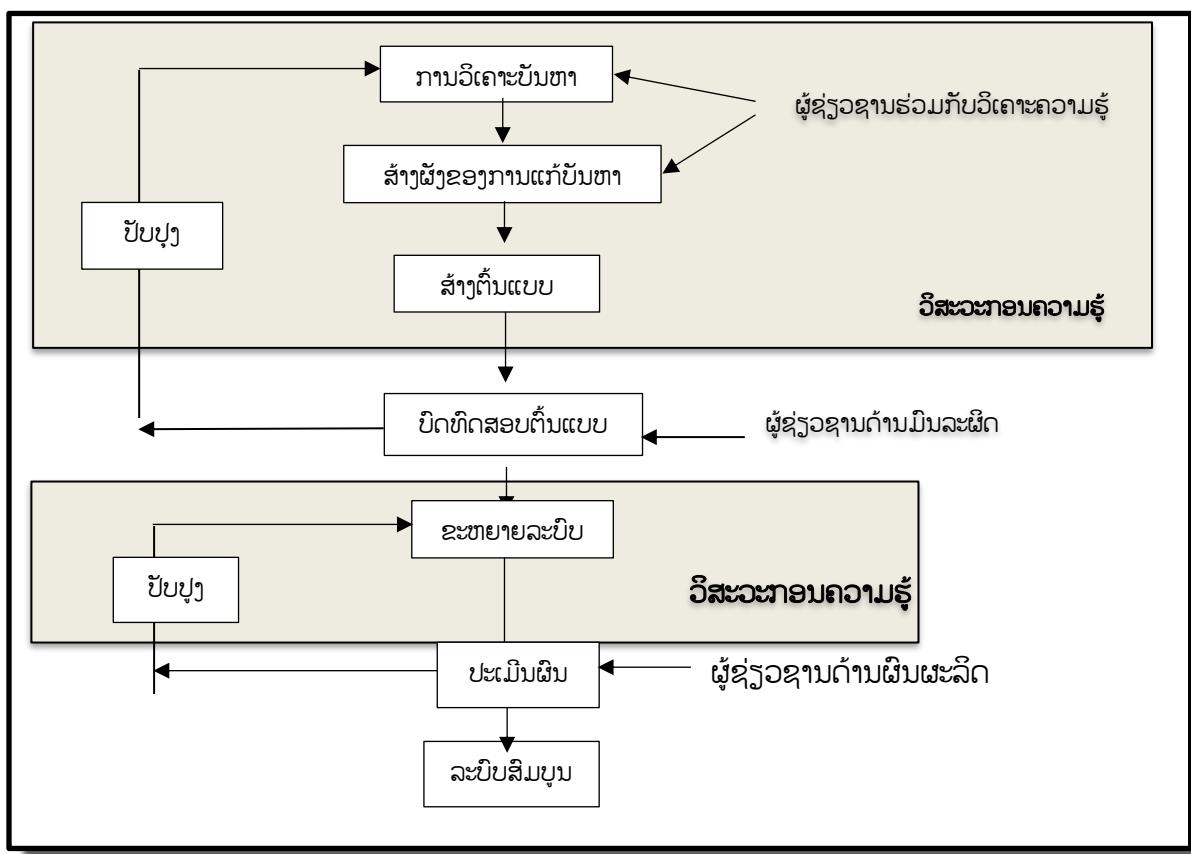
ສໍາລັບການພັດທະນາລະບົບຜູ້ຂໍ່ວຊານຂະໜາດໃຫ່ຍ ວົງຈອນການພັດທະນາຈະຕ່າງກັບລະບົບທີ່ມີຂະໜາດນ້ອຍໝາຍ ເນື່ອງຈາກຂະໜາດຂອງລະບົບທີ່ໃຫ່ຍຂຶ້ນ ຄວາມສັບຊ້ອນຂອງລະບົບກໍ່ຕ້ອງມີຫຼາຍຂຶ້ນເຊັ້ນກັນ, ຄວາມຜິດພາດທີ່ອາດເກີດຂຶ້ນ ນັ້ນໝາຍເຖິງເວລາ ແລະ ເງິນລົງທຶນທີ່ເພີ່ມຂຶ້ນຈຳນວນມະຫາສານ. ດັ່ງນັ້ນ, ການພັດທະນາລະບົບຜູ້ຂໍ່ວຊານຂະໜາດໃຫ່ຍຈົ່ງຕ້ອງມີການວາງແຜນຢ່າງລະມັດລະວັງ ເພື່ອຫຼຸດຜ່ອນຄວາມຜິດພາດທີ່ອາດຈະເກີດຂຶ້ນໃຫ້ໜ້ອຍທີ່ສຸດ.

ໃນຮູບທີ 8.1 ສະແດງວົງຈອນຂອງການພັດທະນາລະບົບຜູ້ຂໍ່ວຊານ ເຊິ່ງຈະເຫັນວ່າໃນການພັດທະນາລະບົບທັງສອງນັ້ນການເຂົ້າຮ່ວມຂອງຜູ້ຂໍ່ວຊານໃນການພັດທະນາລະບົບຈະແຕກຕ່າງກັນ.

ສໍາລັບລະບົບຂະໜາດນີ້ຍ ການເຂົ້າຮ່ວມພັດທະນາລະບົບຂອງຜູ້ຊ່ວຊານຈະໜ້ອຍກວ່າໃນລະບົບໃຫ້ຍຜູ້ທີ່ເຮັດວຽກຫຼັກງານແມ່ນວິສາວະກອນຄວາມຮູ້ (knowledge engineering). ສໍາລັບລະບົບຂະໜາດໃຫ້ຍຜູ້ຊ່ວຊານຈະມີສ່ວນຮ່ວມໃນການພັດທະນາໝາຍ, ຜູ້ຊ່ວຊານຈະຕ້ອງອຸທິດເວລາໝາຍທີ່ສຸດໃຫ້ແກ່ການພັດທະນາລະບົບ ແລະ ອາດຈະຕ້ອງເຂົ້າຮ່ວມໃນການພັດທະນາຢ່າງໄກສິດ.

ການພັດທະນາລະບົບຜູ້ຊ່ວຊານຂະໜາດນີ້ຍ ຈະເລີ່ມຕົ້ນດ້ວຍການເລືອກເຄື່ອງມື ແລະ ທຳຄວາມເຂົ້າໃຈຕໍ່ກັບບັນຫາກ່ຽວກັບລັກສະນະການໃຫ້ຄໍາປຶກສາ. ຈາກນັ້ນກໍເຮັດການຈຳແນກບັນຫາ ແລະ ວິເຄາະຄວາມຮູ້ທີ່ຈະໃສ່ລົງໄປໃນຖານຄວາມຮູ້, ແລ້ວອອກແບບ ແລະ ສ້າງລະບົບຕົ້ນແບບ ໂດຍການກຳນົດຂອບເຂດຄວາມຮູ້ໃຫ້ແບບລົງ. ແລ້ວຂະໜາຍໝາຍ, ຫິດສອບ ແລະ ບັບປຸງລະບົບຈົນກວ່າຈະໃຊ້ໄດ້. ຖ້າຫາກວ່າມີບັນຫາເກີດຂຶ້ນ ເຮົາກໍຈະຍົນກັບໄປທີ່ການຈຳແນກບັນຫາ ແລະ ວິເຄາະຄວາມຮູ້ໃໝ່ ຈົນກວ່າລະບົບຈະເປັນໄປຕາມທີ່ເຮົາຕ້ອງການ.

ຈາກແຜນຜັງຂອງວົງຈອນການພັດທະນາລະບົບຜູ້ຊ່ວຊານຂະໜາດໃຫ້ຍ ຈະເຫັນວ່າຜູ້ທີ່ມີບົດບາດໃນການພັດທະນາລະບົບແມ່ນວິສາວະກອນຄວາມຮູ້ ແລະ ຜູ້ຊ່ວຊານ. ການພັດທະນາລະບົບສ່ວນໃຫ້ຍ ຈະຕ້ອງອາໄສການປະສານງານຂອງທັງວິສາວະກອນຄວາມຮູ້ ແລະ ຜູ້ຊ່ວຊານ. ວົງຈອນການພັດທະນາ ຈະເລີ່ມຕົ້ນຈາກການວິເຄາະບັນຫາ, ເລືອກເຄື່ອງມື, ຈາກນັ້ນກໍຈະເປັນການສ້າງລະບົບຕົ້ນແບບ ແລ້ວກໍເຮັດການຫິດສອບ ຖ້າຫາກວ່າລະບົບຕົ້ນແບບທີ່ໄດ້ຍັງບໍ່ຖືກຕ້ອງ, ກໍ່ຕ້ອງກັບໄປບັບປຸງລະບົບໃໝ່ ໂດຍເລີ່ມຈາກການວິເຄາະບັນຫາ, ບັບປຸງລະບົບຕົ້ນແບບ ແລະ ຫິດສອບໃໝ່ຈົນກວ່າຈະໄດ້ຕົ້ນແບບທີ່ຖືກຕ້ອງ.



ຮູບທີ 8.1 ວົງຈອນການພັດທະນາລະບົບຜູ້ຂ່າວຊານຂະໜາດໃຫ້ຍ

ເມື່ອໄດ້ຕົ້ນແບບທີ່ຖືກຕ້ອງແລ້ວ, ເຮັດການຂະຫຍາຍລະບົບໃຫ້ເປັນລະບົບທີ່ສົມບູນ ແລະ ປະເມີນຜົນ. ຖ້າລະບົບທີ່ຂະຫຍາຍຂຶ້ນມາມີຫຍັງຕ້ອງແກ້ໄຂກໍຈະຕ້ອງກັບໄປສ້າງລະບົບທີ່ສົມບູນໃໝ່, ແລ້ວເຮັດການປະເມີນຜົນໃໝ່ຈິນກວ່າຈະໄດ້ຜົນທີ່ພື້ນໃຈ ເມື່ອໄດ້ລະບົບທີ່ພື້ນໃຈແລ້ວ ອໍາທິດຕັ້ງລະບົບ ແລະ ວາງແຜນການບໍາລຸງຮັກສາເພື່ອເຮັດໃຫ້ລະບົບຜູ້ຂ່າວຊານນີ້ມີຄວາມຮູ້ທີ່ທັນສະໄໝ. ລາຍລະອຽດຂັ້ນຕອນຕ່າງໆຂອງການພັດທະນາລະບົບຈະໄດ້ອະນຸມາຍຕັ້ງຕໍ່ໄປນີ້:

8.1.1. ການຈຳແນກປັນຫາ ແລະ ວິເຄາະຄວາມຮູ້ທີ່ຈະສະຫຼຸບໄສ່ຖານຄວາມຮູ້

ຄວາມແຕກຕ່າງຂອງການພັດທະນາລະບົບທີ່ມີຂະໜາດນັ້ນ ແລະ ຂະໜາດໃຫ້ຍອີກຍ່າງໜຶ່ງກໍຄື ໃນລະບົບຂະໜາດນັ້ນຍັນ ວົງຈອນການພັດທະນາຈະເລີ່ມຕົ້ນດ້ວຍການເລືອກເຄື່ອງມື, ແຕ່ສໍາລັບການພັດທະນາລະບົບທີ່ມີຂະໜາດໃຫ້ຍ ວົງຈອນການພັດທະນາຈະເລີ່ມຈາກການວິເຄາະປັນຫາ, ເພາະໃນການພັດທະນາລະບົບທີ່ມີຂະໜາດນັ້ນຍັນຈະສາມາດເຮັດໄດ້ໂດຍການເອົາເຄື່ອງມືທີ່ມີຢູ່ມາໃຊ້ ແລະ ນຳຄວາມຮູ້ທີ່ຈະພັດທະນາມາວິເຄາະເພື່ອໃຊ້ກັບເຄື່ອງມືທີ່ມີຢູ່ແລ້ວ ຍັນວ່າຄວາມຮູ້ຂະໜາດນັ້ນຍັນບໍ່ມີຄວາມສັບຊັອນໜ້າຍ.

ສໍາລັບລະບົບທີ່ມີຂະໜາດໃຫ້ຍ ການພັດທະນາຄວາມຮູ້ຈະຕ້ອງລົງທຶນສູງ ຜູ້ພັດທະນາຈະຕ້ອງພົມທີ່ຈະລົງທຶນຊື້ເຄື່ອງມືໃໝ່ເພາະຄວາມສັບຊັອນຂອງລະບົບ. ດັ່ງນັ້ນວົງຈອນການພັດທະນາຈະຕ້ອງເລີ່ມຕົ້ນດ້ວຍການວິເຄາະປັນຫາ, ແລ້ວຈຶງເລືອກເຄື່ອງມືທີ່ເໝາະສົມ, ເພາະປັນຫາທີ່ຕ່າງກັນຈະເໝາະກັບເຄື່ອງມືທີ່ຕ່າງກັນ.

ການພັດລະບົບຜູ້ຂ່າວຊານຂະໜາດໃຫ້ຍ ຍັງມີເລື່ອງທີ່ຄວນລະວັງອີກຫຼາຍຢ່າງເຊື່ອງຈະໄດ້ກ່າວຕໍ່ໄປຢ່າງລະອຽດໃນຫົວຂໍ້ເລື່ອງການເລືອກປັນຫາທີ່ເໝາະສົມສໍາລັບການພັດທະນາລະບົບຜູ້ຂ່າວຊານຂະໜາດໃຫ້ຍ.

ໃນການພັດລະບົບທີ່ມີຂະໜາດນັ້ນ ເມື່ອໄດ້ເລືອກເຄື່ອງມືແລ້ວຈະຮູ້ວ່າເຄື່ອງມືທີ່ເລືອກໃຊ້ນັ້ນມີລັກສະນະການໃຫ້ຄໍາບຶກສາ, ການສະແດງຄວາມຮູ້ ແລະ ຂອບເຂດຂອງປັນຫາເປັນແນວໃດ. ດັ່ງນັ້ນ, ສີ່ງທີ່ຈະຕ້ອງເຮັດຕໍ່ໄປຄືການຈຳແນກປັນຫາ ແລະ ວິເຄາະຄວາມຮູ້, ລັກສະນະການເຮັດວຽກ ແລະ ການວິເຄາະປັນຫາຂອງລະບົບຜູ້ຂ່າວຊານມີຫຼາຍວິທີເຊື່ອງຈະໄດ້ກ່າວເຖິງຕາມຫຼັງ.

ການແກ້ໄຂປັນຫາໃນລະບົບຜູ້ຂ່າວຊານມີລັກສະນະຄ້າຍຕືກກັບການໃຫ້ຄໍາບຶກສາທາງໂທລະສັບ ຖ້າສົມມຸດວ່າ ຫ້າວ ກ ກຳລັງເຈັບໄສ້ຕື່ງຢ່າງຮຸນແຮງຕ້ອງໄດ້ຮັບການຜ່າຕັດຢ່າງຮີບດ່ວນ, ຢູ່ເຮືອນຫ້າວ ກ ມີຫ້າວ ຂ ຢູ່ຄົນດູວ, ເຊິ່ງບໍ່ມີຄວາມຮູ້ທາງດ້ານການຜ່າຕັດເລີຍ. ຫ້າວ ຂ ໂທລະສັບຫາ ຫ້າວ ຄ ຕີ່ເປັນຫ່ານໝໍຂ່າວຊານດ້ານການຜ່າຕັດໄສ້ຕື່ງວ່າຄວນຈະເຮັດແນວໃດ, ຂະນະນີ້ ຫ້າວ ຄ ແມ່ນຜູ້ຂ່າວຊານທີ່ຈະຕ້ອງຕັ້ງຄໍາຖາມ ເພື່ອຖາມ ຫ້າວ ຂ ກ່ຽວກັບອາການຂອງ ຫ້າວ ກ, ເມື່ອໄດ້ຂັ້ນມູນບາງຢ່າງກໍສາມາດແນຍະນຳ ຫ້າວ ຂ ໃຫ້ປະຕິບັດຕາມຂັ້ນຕອນຂອງການຜ່າຕັດໄສ້ຕື່ງໄດ້. ການຈຳແນກປັນຫາຂອງຫ້າວ ຄ ກໍເຊັ້ນດູວກັບການຈຳແນກປັນຫາເພື່ອເອົາມາໃສ່ໃນຖານຄວາມຮູ້ ຫ້າຫາກວ່າຫ່ານຈະຈຳແນກ

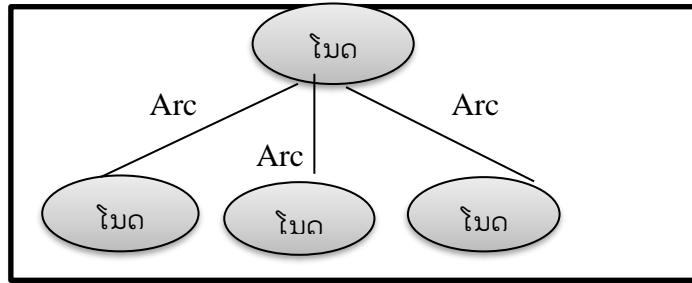
1. ການມີຄວາມຮູ້ ແລະ ຄວາມເຂົ້າໃຈປັນຫາຢ່າງຖືກຕ້ອງ ເປັນພື້ນຖານສໍາລັບການພັດທະນາລະບິບ
ຄວາມຮູ້, ເຊັ່ນ: ຖ້າຈະສ້າງຖານຄວາມຮູ້ກ່ຽວກັບການຖ່າຍຮູບ ກໍ່ຕ້ອງມີຄວາມຮູ້ກ່ຽວກັບການ
ຖ່າຍຮູບ ແລະ ຈະຕ້ອງເຂົ້າໃຈວ່າໃນຂະບວນການຖ່າຍຮູບນັ້ນມີປັນຫາຫຍ້ງແດ່. ເງື່ອນໄຂຕ່າງໆ
ໃນຄວາມຮູ້ເລື່ອງນັ້ນມີວິງປະກອບຫຍ້າທີ່ມີບິດບາດສໍາຄັນ, ເຊັ່ນ: ແສງທີ່ຕ່າງກັນ, ການຕັ້ງຄວາມ
ໄວກໍຕ່າງກັນໃນເງື່ອນໄຂຂອງພິມ (asa) ຕ່າງກັນ ແລະ ການຕັ້ງໜ້າກ້ອງກໍຕ່າງກັນພ້ອມ.

ពិវឌ្ឍន៍ទី 8.1 ទំនាក់ទំនងក្នុងការពារតាមនូវ

asa	Light-condition	distance	aperture	
100	bright-sun	< 12 > 12	f11	Speed 125
	soft-shadow		f8	Speed 125
	cloudy-bright		f5.6	Speed 125
	over-case		f4	Speed 125
	flash		f8	Speed 125
200	bright-sun	< 12 > 12	f16	Speed 250
	soft-shadow		f8	Speed 250
	cloudy-bright		f5.6	Speed 250
	over-case		f4	Speed 250
	flash		f11	Speed 250
400	bright-sun	< 12 > 12	f8	Speed 250
	soft-shadow		f16	Speed 500
	cloudy-bright		f11	Speed 500
	over-case		f5.6	Speed 500
	flash		f5.6	Speed 250

2. ภารกิจจัดขั้นตอนเพื่อแก้ปัญหา ผู้ช่วยฯ วางแผนต้องมีความเข้าใจว่าจะบวบเวียนของภารกิจที่มีอยู่แล้วเป็นแบบใด มีวิธีการແນວໃใจในการแก้ปัญหา ขั้นตอนทุกขั้นตอนจะต้องมีภารกิจจัดลำดับ และ กำหนดวิธีการแก้ปัญหา ดังต่อไปนี้ ได้แก่ วิธีการจัดขั้นตอนในภารกิจแก้ปัญหาโดยอาศัยต้นตระหง่าน อิกรูปแบบหนึ่งที่สามารถอธิบายในภารกิจจัดขั้นตอนในภารกิจแก้ปัญหาได้ดีคือ ภารกิจจัดความรู้ในรูปแบบของต้นไม้ (tree) ใบกำลังมีที่ความรู้มีขั้นขนาดใหญ่ และ สับข้อน้อย ภารกิจให้รู้วิธีการจำแนกปัญหาในตัวอย่างที่ 8.1 คาดจะขยายไปยังภารกิจแก้ปัญหาที่มีความซับซ้อนมากขึ้น ภารกิจจะแบ่งเป็นวิธีสังเคราะห์ ภารกิจจะแบ่งเป็นวิธีสังเคราะห์ ภารกิจจะแบ่งเป็นวิธีสังเคราะห์

ໃນຖານຄວາມຮູ້ ແລະ ຈົ່າຈະເປັນສ່ວນເຊື່ອມຄວາມສຳພັນຂອງໄໂນດຕ່າງໆ ທີ່ກ່ຽວຂ້ອງກັນເຂົ້າກັນ.

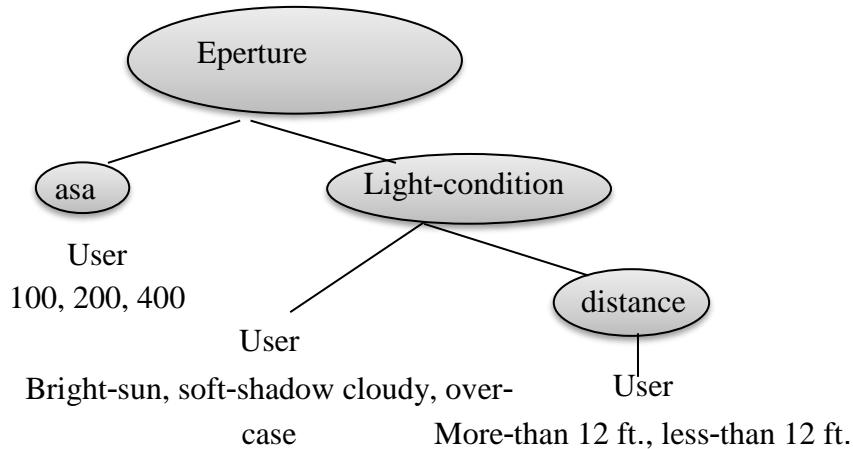


ຮູບທີ 8.2 ໂຄງສ້າງຕົ້ນໄມ້

ໃນສ່ວນຂອງຄວາມສຳພັນລະຫວ່າງໄໂນດ ຈະກຳນົດໃຫ້ໄໂນດຕົວທີ່ຢູ່ໃນລະດັບທີ່ສູງກວ່າເປັນ attribute ຂອງໄໂນດລຸ່ມ ແລະ ໄໂນດລຸ່ມຈະເປັນຄ່າຂອງໄໂນດເທິງ ໂດຍມີງາເຊື່ອມຄວາມສຳພັນຂອງໄໂນດທີ່ເປັນ attribute ແລະ ຄ່າເຂົ້າກັນ ເຊິ່ງຈະກຳນົດຄ່າຄວາມສຳພັນໃຫ້ກັບງ່າງ ຫຼືບໍ່ກໍໄດ້.

3. ການຖາມຄໍາຖາມເພື່ອເປັນຂໍ້ມູນວ່າຄໍາຕອບຈະເປັນແບບໄດ້ ປັນຫາສະເພາະໜຶ່ງຈະມີວິທີການແກ້ປັນຫາສະເພາະຢ່າງ, ຖ້າປັນຫາເປັນແບບໜຶ່ງການແກ້ປັນຫາຈະເປັນແບບໜຶ່ງ ໃນປັນຫາດຽວກັນຈະມີວິທີການແກ້ປັນຫາໄດ້ຫຼາຍຢ່າງ ແລະ ວິທີການສະເພາະນີ້ຈະຂຶ້ນຢູ່ກັບລັກສະນະສະເພາະຂອງປັນຫາ. ໃນການທີ່ຈະໄດ້ມາເຊິ່ງລັກສະນະສະເພາະຂອງປັນຫາ ເມື່ອໄດ້ລັກສະນະສະເພາະຂອງປັນຫາກໍສາມາດໃຫ້ຄໍາປິກສາໃນການແກ້ປັນຫານັ້ນໄດ້ຢ່າງຖືກຕ້ອງ, ການຕັ້ງຄໍາຖາມຕ້ອງງ່າຍຕໍ່ຄວາມເຂົ້າໃຈ ແລະ ເຮັດໃຫ້ຜູ້ໃຊ້ສາມາດຕອບຄໍາຖາມໄດ້ຢ່າງຕົງປະເດັນ. ຈາກຕົວຢ່າງທີ 8.2 ໃນສ່ວນຂອງຜູ້ໃຊ້ຄືສ່ວນທີ່ລະບົບຈະຕ້ອງຕິດຕໍ່ກັບຜູ້ໃຊ້ ເຊິ່ງຜູ້ທີ່ສ້າງຖານຄວາມຮູ້ຈະຕ້ອງຕັ້ງເປັນຄໍາຖາມທີ່ຈະຕິດຕໍ່ກັບຜູ້ໃຊ້.

ຕົວຢ່າງທີ 8.2 ໂຄງສ້າງແບບຕົ້ນໄມ້ຂອງຖານຄວາມຮູ້ເລື່ອງນັກທ່າຍຮູບມືໃໝ່



ຮູບທີ 8.3 ໂຄງສ້າງຕົ້ນໄມ້ຂອງຖານຄວາມຮູ້

4. ການໃຫ້ຄໍາປຶກສາ ຈະຕ້ອງເປັນລັກສະນະທີ່ເຂົ້າໃຈ ແລະ ສາມາດປະຕິບັດຕາມໄດ້ງ່າຍ ການຕັ້ງຄໍາຖານຈະຕັ້ງຈະແຈ້ງ ໃຫ້ຜູ້ໃຊ້ສາມາດຕອບຄໍາຖານໄດ້ຢ່າງຕົງປະເດັນ.

ຕົວຢ່າງທີ 8.3 ຈາກຕົວຢ່າງທີ 8.1 ແລະ ຕົວຢ່າງທີ 8.2 ຈະເຫັນໄດ້ວ່າຄໍາຖານທີ່ລະບົບຜູ້ຂ່າງວານຕ້ອງການຢາກຮູ້ເພື່ອໃຊ້ໃນການຕັດສິນໃຈຄື:

- ກ່ຽວກັບເງື່ອນໄຂຂອງແສງ (Light-condition)
- ຄວາມໄວແສງຂອງຟິມ (film-speed)
- ໄລຍະຫ່າງ

8.1.2. ການເລືອກເຕື່ອງມື ແລະ ສ້າງຄວາມເຂົ້າໃຈກ່ຽວກັບລັກສະນະຂອງການໃຫ້ຄໍາປຶກສາ

ບັດຈຸບັນເຕື່ອງມືທີ່ໃຊ້ໃນການພັດທະນາລະບົບຄວາມຮູ້ມີຫຼາຍຊະນິດ ແລະ ແຕ່ລະຊະນິດກໍມີວິທີການສະແດງຄວາມຮູ້ທີ່ແຕກຕ່າງກັນໄປ. ດັ່ງນັ້ນໃນການເລືອກເຕື່ອງມືແຕ່ລະຊະນິດຈໍາເປັນຈະຕັ້ງເລືອກໃຫ້ເໝາະຮັບວຽກທີ່ຈະເຮັດ. ນອກຈາກທີ່ກ່າວມາແລວຍ້າມີລັກສະນະພິເສດຂອງເຕື່ອງມືແຕ່ລະຊະນິດອີກດ້ວຍ, ເຊັ່ນ: ຄວາມສາມາດໃນການບັນຈຸຄວາມຮູ້, ຄວາມສາມາດໃນການສະແດງພາບ, ສູງ ແລະ ການຕິດຕໍ່ຮັບອຸປະກອນພາຍນອກ ເປັນຕົ້ນ.

ນອກຈາກເຕື່ອງມືທີ່ເຕີມໃຈພິຈາລະນາແລ້ວ ໃນສ່ວນຂອງປັນຫາແມ່ນຈໍາເປັນທີ່ສຸດທີ່ຈະຕັ້ງພິຈາລະນາເຖິງຂອບເຂດຂອງການໃຫ້ຄໍາປຶກສາ (domain) ພ້ອມ. ສໍາລັບການພັດທະນາລະບົບຜູ້ຂ່າງວານທີ່ມີຂະໜາດນ້ອຍ ພໍມີວິທີຄ້າວ່າງທີ່ໃຊ້ເປັນຫຼັກເຕັມໃນການພິຈາລະນາວ່າລະບົບນີ້ເປັນລະບົບທີ່ມີຂອບເຂດຂະໜາດນ້ອຍໄດ້ດັ່ງຕໍ່ໄປນີ້:

- ເວລາທີ່ໃຊ້ໃນການໃຫ້ຄໍາປຶກສາທັງໝົດບໍ່ຄວນເກີນ 30 ນາທີ ນັບຕັ້ງແຕ່ເລີ່ມການຖາມ-ຕອບຄໍາຖາມ ຈຶນເຖິງຄໍາແນະນຳຂັ້ນສຸດທ້າຍ.
- ຄວາມເປັນໄປໄດ້ຂອງຄໍາຕອບທີ່ລະບົບຜູ້ຂ່າງວານຈະຕັ້ງເລືອກບໍ່ຄວນເກີນ 50 ຊຸດ.

ຕົວຢ່າງ 8.4: ການພັດທະນາລະບົບຄວາມຮູ້ທີ່ໃຫ້ຄໍາປຶກສາກ່ຽວກັບການຖ່າຍຮູບ ໂດຍທີ່ວ່າໄປແລ້ວເຕີມຈໍາເປັນທີ່ຈະຕັ້ງອ່າໄສການສະແດງພາບປະກອບ ແລະ ການສະແດງຄວາມຮູ້ກໍສາມາດໃຊ້ໄດ້ດີກັບ

ການສະແດງໂດຍຮູບແບບຂອງກົດ ເພົ່າຈຳເປັນຕົວອາໄສການຄໍານວນເຂົ້າມາປະກອບ ຈາກຕົວຢ່າງຊ່າງຖ່າຍຮູບມີໃໝ່ເຮົາຈະທິດລອງໃຊ້ M.1 ເຊິ່ງເປັນລະບົບຜູ້ຂ່າຍຊານຊະນິດ shell ທີ່ເປັນການອະນຸມານແບບກັບຫຼັງ ມີວິທີການສະແດງຄວາມຮູ້ໂດຍອາໄສກົດ ແລະ ມີລັກສະນະການໃຫ້ຄໍາປຶກສາແບບຖາມ-ຕອບທີ່ສົມບູນທີ່ສຸດລະບົບໜີ່ງ. ນອກຈາກນັ້ນແລ້ວ M.1 ຍັງເປັນລະບົບຜູ້ຂ່າຍຊານຊະນິດ shell ທີ່ສາມາດໃຊ້ໄດ້ກັບການພັດທະນາລະບົບທີ່ມີຂະໜາດນັ້ນອຍ ແລະ ສາມາດໃຊ້ວິກັບເຄື່ອງຄອມພິວເຕີບຸກຄົນ (PC: Personal Computer) ໄດ້.

8.1.3. ການອອກແບບ

ການອອກແບບລະບົບຄວນເລີ່ມຕົ້ນຈາກເຈັຍ ຂູນແວຄວາມຄົດຂອງຄວາມຮູ້ຫັງໝີດທີ່ເຮົາຈະສ້າງໂດຍເລີ່ມຕົ້ນຈາກ:

- ຈະຕົວມີເປົ້າໝາຍທີ່ຈະແຈ້ງ, ເປົ້າໝາຍນີ້ໝາຍເຖິງຈຸດໝາຍປາຍທາງຂອງລະບົບວ່າການໃຫ້ຄໍາປຶກສານີ້ຈະເປັນແວວິດ ຫຼືເວົາງ່າຍ່າງກໍ່ຄື ຄໍາຕອບຂອງການໃຫ້ຄໍາປຶກສານັ້ນເອງ ແລະ ຄໍາຕອບນີ້ຈະມີຢູ່ໝາຍງົງຄໍາຕອບ ເຊິ່ງລະບົບຜູ້ຂ່າຍຊານຈະເປັນຜູ້ເລືອກໃຫ້ສອດຄ່ອງກັບລັກສະນະສະເພາະຂອງປັນຫາ. ຈາກຕົວຢ່າງເລື່ອງ: “ຊ່າງຖ່າຍຮູບມີໃໝ່” ຈະສາມາດກຳນົດໄດ້ວ່າ:
ຈາກໂຄງສ້າງຕົ້ນໄມ້ໃນຕົວຢ່າງທີ 8.2 ເລື່ອງຊ່າງຖ່າຍຮູບມີໃໝ່ ເປົ້າໝາຍຂອງຄວາມຮູ້ຈະຢູ່ທີ່ໂນດເຫິງສຸດ ເຊິ່ງກໍ່ຄື aperture. ດັ່ງນັ້ນ goal = aperture.
- ເງື່ອນໄຂຂອງແສງ
- ເງື່ອນໄຂຂອງ asa
- ໄລຍະຫ່າງ (ຖ້າໃຊ້ເງື່ອນໄຂຂອງແສງ = ໃຊ້ flash)
- ໃຫ້ຄໍາປຶກສາ ດັ່ງຕາຕະລາງທີ່ສະແດງຄວາມຮູ້ເບື້ອງຕົ້ນທີ່ໄດ້ກ່າວໄປແລ້ວ

8.1.4 ການສ້າງຕົ້ນແບບ

ໃນການສ້າງຕົ້ນແບບ (Prototype) ເປັນການສະແດງຄວາມຮູ້ສະເພາະຕອນຂຶ້ນມາ ໂດຍການຈຳກັດຂອບເຂດຂອງຄວາມຮູ້ໃຫ້ແຄບລົງ. ໃນການສ້າງລະບົບຕົ້ນແບບນັ້ນ ມີຈຸດປະສົງເພື່ອຫາຄວາມເປັນໄປໄດ້ຂອງການສ້າງລະບົບ ແລະ ຫາທິນຫາງໃນການແກ້ປັນຫາກ່ອນທີ່ຈະສ້າງລະບົບແທ້.

ຕົວຢ່າງທີ 8.5 ຈາກຄວາມຮູ້ກ່ຽວກັບການຖ່າຍພາບໃນຕົວຢ່າງທີ 8.1 ທິດສອບເງື່ອນໄຂ light-condition ເທິງກັບ bright-sun ແລະ asa = 100, 200 ຈະໄດ້ຕົ້ນແບບອອກມາດັ່ງນີ້:

goal = advice

1. if light-condition = bright-sun and asa = 100 then

advice = ‘aperture = f11 and shutter-speed = 125’

2. if light-condition = bright-sun and asa = 200 then

advice = ‘aperture = f16 and shutter-speed = 250’

question (light-condition) = “What is the condition of light?”

question (asa) = ‘What is the condition of light?’

ລະບົບຕົ້ນແບບທີ່ສ້າງຂຶ້ນນີ້ ຈະຕ້ອງມີລັກສະນະການເຮັດວຽກທີ່ຄືກັບລະບົບຈິງທີ່ຈະພັດທະນາຕໍ່ແຕ່ກໍານົດໃຫ້ຂອບເຂດຂອງການແກ້ໄຂບັນຫາເຮັດໄດ້ໜ້ອຍກວ່າ. ລະບົບຕົ້ນແບບນີ້ຈະເປັນຕົ້ນແບບເພື່ອໃຊ້ໃນການທິດສອບວ່າການແກ້ບັນຫາທີ່ໄດ້ເຮັດການອອກແບບມານັ້ນຖືກຕ້ອງ ຫຼື່ນໍ້າ ແລະ ເພື່ອເປັນແນວທາງໃນການຂະໜາຍລະບົບຕໍ່ໄປ.

8.1.5 ການຂະໜາຍ, ທິດສອບ ແລະ ການປັບປຸງລະບົບ

ການຂະໜາຍລະບົບໂດຍການເອົາຕົ້ນແບບທີ່ແນ່ໃຈວ່າຖືກຕ້ອງແລ້ວມາເຮັດການເພີ່ມອົງປະກອບຕ່າງໆ ຈົນເປັນລະບົບທີ່ສົມບູນຕາມທີ່ໄດ້ການວາງແຜນໄວ້ ໂດຍການເພີ່ມຄວາມຮູ້ໃນສ່ວນທີ່ຍັງຈາດຢູ່, ຕ້ອງຕົກແຕ່ງລະບົບໃຫ້ເບິ່ງເປັນແບບບານິດ ແລະ ເພີ່ມສ່ວນທີ່ໃຊ້ໃນການອະທິບາຍສ່ວນຕ່າງໆ.

ໃນການພັດທະນາລະບົບຜູ້ຂ່າວຊານຂະໜາດໃຫ່ຍ ກ່ອນທີ່ຈະມີການຂະໜາຍລະບົບຕົ້ນແບບນີ້ ຈະຕ້ອງມີການກວດສອບໂດຍຜູ້ຂ່າວຊານ ແລະ ວິສະວະກອນຄວາມຮູ້ຢ່າງລະອຽດ ໂດຍການເອົາເງື່ອນໄຂຕ່າງໆທີ່ໄດ້ວາງໄວ້ໃນການສ້າງລະບົບຕົ້ນແບບມາເຮັດການທິດສອບ ແລະ ກວດໂດຍຜູ້ຂ່າວຊານ ເພື່ອເບິ່ງວ່າເງື່ອນໄຂທິດສອບນັ້ນຖືກຕ້ອງ ຫຼື່ນໍ້າ. ຖ້າທາກວ່າລະບົບຕົ້ນແບບມີຄວາມຄາດເລື່ອນຈາກລະບົບທີ່ວາງເອົາໄວ້ກໍຈະຕ້ອງກັບໄປເຮັດການອອກແບບລະບົບຕົ້ນແບບໃໝ່ ສໍາຫຼັບການທິດສອບລະບົບຕົ້ນແບບມີສິ່ງທີ່ຕ້ອງຄໍານິ່ງສະເໝີວ່າລະບົບນີ້ໄດ້ມີການຈຳລອງລະບົບໃຫ້ມີຂອບເຂດຂອງການແກ້ບັນຫາທີ່ໜ້ອຍລົງກວ່າລະບົບຈິງ. ດັ່ງນັ້ນເງື່ອນໄຂໃນການທິດສອບບາງຢ່າງທີ່ບໍ່ໄດ້ກໍານົດໄວ້ໃນການສ້າງລະບົບຕົ້ນແບບກໍຈະນຳມາກວດສອບບໍ່ໄດ້.

ສ່ວນການປະເມີນຜົນຂອງລະບົບເມື່ອລະບົບສ້າງສໍາເລັດແລ້ວ ຈະຕ້ອງມີການປະເມີນຜົນວ່າລະບົບທີ່ໄດ້ອອກແບບມານີ້ເປັນໄປຕາມຄວາມຕ້ອງການຂອງຜູ້ອອກແບບລະບົບ ຫຼື່ນໍ້າ? ໃນການກວດສອບຜູ້ກວດສອບຈະຕ້ອງມີຜູ້ຂ່າວຊານທີ່ມາຊ່ວຍໃນການພັດທະນາ ມາໃຫ້ຄໍາປຶກສາຢ່າງໃກ້ຊີດ, ວິສະວະກອນຄວາມຮູ້ຈະຕ້ອງກວດສອບເງື່ອນໄຂຕ່າງໆຂອງການອະນຸມານໃຫ້ຄົບຖວນ ແລະ ຜູ້ຂ່າວຊານຈະຕ້ອງກວດສອບຄວາມຮູ້ທຸກຢ່າງທີ່ມີຢູ່ໃນລະບົບວ່າກົງກັບຄວາມເປັນຈີງ ຫຼື່ນໍ້າ? ຖ້າທາກວ່າເກີດຄວາມຜິດພາດວິສະວະກອນຄວາມຮູ້ຈະຕ້ອງເປັນຜູ້ແກ້ໄຂກົດ ຫຼື້ຂັ້ນມູນຕ່າງໆໃນຖານຄວາມຮູ້.

ການປຳລຸງຮັກສາລະບົບຜູ້ຂ່າວຊານ, ຜູ້ສ້າງຈະຕ້ອງເຂົ້າໃຈວ່າຄວາມຮູ້ທີ່ໄສ່ເຂົ້າໄປໃນລະບົບຜູ້ຂ່າວຊານນັ້ນຫຼົາສະ ໄໝ ແລະ ປົງແປງໄດ້ ຈຳເປັນຈະຕ້ອງເພີ່ມເຕີມໃນອະນາຄົດ. ດັ່ງນັ້ນ, ລະບົບຜູ້ຂ່າວຊານທີ່ສ້າງຂຶ້ນໃນລະບົບໃຫຍ່ຈຶ່ງຈຳເປັນຫຼາຍທີ່ຈະຕ້ອງມີການເພີ່ມເຕີມຄວາມຮູ້ ຫຼື ຈະຕ້ອງມີການປັບປຸງແກ້ໄຂຢ່າງສະໜໍ່ສະເໝີ ເພື່ອຮັດໃຫ້ລະບົບຜູ້ຂ່າວຊານນີ້ມີຄວາມທັນສະໄໝຢູ່ຕະຫຼອດ.

8.2. ການເລືອກບັນຫາໃຫ້ເໝາະສົມສໍາລັບການພັດທະນາລະບົບຜູ້ຂ່າວຊານຂະໜາດໃຫ່ຍ ມີສິ່ງທີ່ໜ້າສົນໃຈດັ່ງນີ້:

1. ຈຳແນກໂດເມນຂອງບັນຫາ ແລະ ບັນຫາສະເພາະຂອງວຽກ: ໃນການເລືອກບັນຫາທີ່ຖືກຕ້ອງບາງຄັງ ອາດຈະເປັນສ່ວນທີ່ສໍາຄັນທີ່ສຸດຂອງການພັດທະນາລະບົບ. ໃນຂັ້ນຕອນນີ້ເຮົາຈະຕ້ອງເຂົ້າໃຈວ່າເຕັກໂນໂລຢີທາງດ້ານນີ້ຢູ່ມີຂໍຈຳກັດຫຼາຍ, ຖ້າທາການເລືອກບັນຫາບໍ່ຖືກຕ້ອງບາງເຖິ່ງການພັດທະນາລະບົບຜູ້ຂ່າວຊານຈະເກີດບັນຫາຂຶ້ນໄດ້ທີ່ໃຜກໍບໍ່ສາມາດຊ່ວຍແກ້ໄຂບັນຫາໄດ້. ດັ່ງນັ້ນ, ການເລືອກ

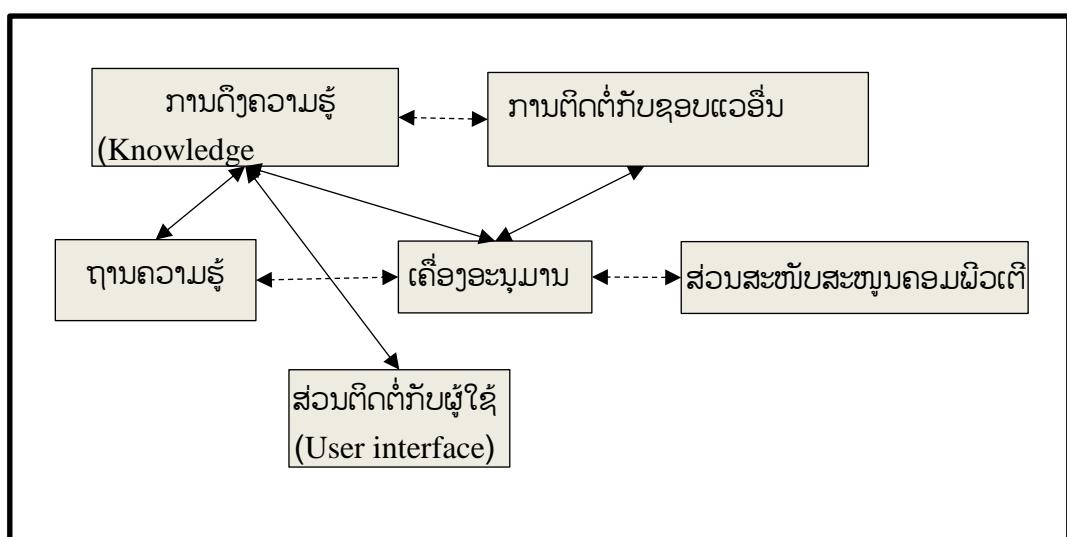
8.2.1. ການຈຳແນກຊະນິດຂອງເຄື່ອງມືທີ່ໃຊ້ໃນການພັດທະນາລະບົບຜູ້ຊ່າວຊານ

ການພັດທະນາລະບົບຜູ້ຊ່ວຂານໃນປັດຈຸບັນມີການປິດແປງຢ່າງວ່ອງໄວທັງໃນແງ່ຂອງຄວາມ່າຍ ແລະຄວາມໄວທີ່ໃຊ້ໃນການພັດທະນາລະບົບ ເນື້ອງຈາກການພັດທະນາຢ່າງວ່ອງໄວຂອງເຄື່ອງມືທີ່ໃຊ້ສໍາຫັບການພັດທະນາລະບົບຜູ້ຊ່ວຂານ (Expert System Building Tool : ESBT) ນັ້ນເອງ. ເຄື່ອງມືເຫຼົ່ານີ້ໂດຍສ່ວນຫຼາຍແລ້ວມັກຈະຖືກສ້າງຂຶ້ນມາເພື່ອການຄ້າ ແລະ ມີລັກສະນະພິເສດໃນການນຳໃຊ້ທີ່ຕ່າງກັນຂຶ້ນຢູ່ກັບວິທີການທີ່ໃຊ້ໃນການສ້າງເຄື່ອງມືເຫຼົ່ານີ້. ໃນທີ່ວັນທີ່ໄປຂອງບົດນີ້ຈະອະທິບາຍເຖິງລາຍລະອຽດ ວິທີການຕ່າງໆ ທີ່ໃຊ້ໃນການສ້າງເຄື່ອງມືເພື່ອການພັດທະນາລະບົບຜູ້ຊ່ວຂານ, ເຊັ່ນ: ລາຍລະອຽດໂຄງສ້າງຂອງເຄື່ອງມືສະແດງຄວາມຮູ້, ເຄື່ອງອະນຸມານການຕິດຕໍ່ຜູ້ພັດທະນາລະບົບ, ການຕິດຕໍ່ກັບ End-User ແລະ ພາສາທີ່ໃຊ້ໃນການພັດທະນາເຄື່ອງມື.

8.2.2. ໂຄງສາງຂອງເຄື່ອງມືທີ່ໃຊ້ໃນການພັດທະນາລະບົບຜູ້ຂ່າວຊານ

ເມື່ອເປົ້າສະເພາະສ່ວນຂອງການຕິດຕໍ່ລະຫວ່າງເຄື່ອງມື (ESBT) ກັບຜູ້ພັດທະນາລະແລ້ວ ຈະເຫັນ
ຄວາມຈຳເປັນຂອງການມີລະບົບນີ້ຂຶ້ນມາຄື:

- เพื่อให้กานพัฒนาตามความรู้ที่ต้องกานมีความสอดคล้องกับลักษณะ
 - เพื่อให้ความสามาดส้างส่วนที่รีดหน้าที่ติดต่อกับ End-user เป็นไปตามที่ผู้พัฒนาจะบังต้องกาน
 - เพื่อให้สามาดเพิ่มส่วนที่รีดหน้าที่ในกานควบคุมกานอนุมานเข้าไปได้ ความสามาดที่กว่ามาเป็นสิ่งที่บอภจะดับของเครื่องมือที่ใช้ในกานพัฒนาจะบังผู้ช่วยงานได้เป็นอย่างดี ว่ามีความสามาดสูง หรือไม่. ส่วนนึนของหน้าที่ที่ใช้ติดต่อกับผู้พัฒนาจะบังแล้วในส่วนของกานติดต่อกับเลื่องอื่นยังเป็นสิ่งที่จำเป็นอีกเช่นกันดังนี้:
 - ความสามาดในกานติดต่อกับซอฟต์แวร และ ฐานข้อมูล (Database) อะนิเดื่อง
 - ความสามาดในกานใช้ปะໂຫຍດจากลักษณะบีบคอมพิวเตอร์ความสามาดในกานติดต่อกับเครื่องขยายเสียงและบีบสีสารเป็นต้น



ຮັບທີ 8.4 ໂຄງສ້າງຂອງ ESBT(ເສັ້ນທີບສະແດງສິ່ງທີ່ສໍາພັນກັນໂດຍພື້ນຖານ ແລະເສັ້ນ

❖ ການສະແດງຄວາມຮັບຮັກ

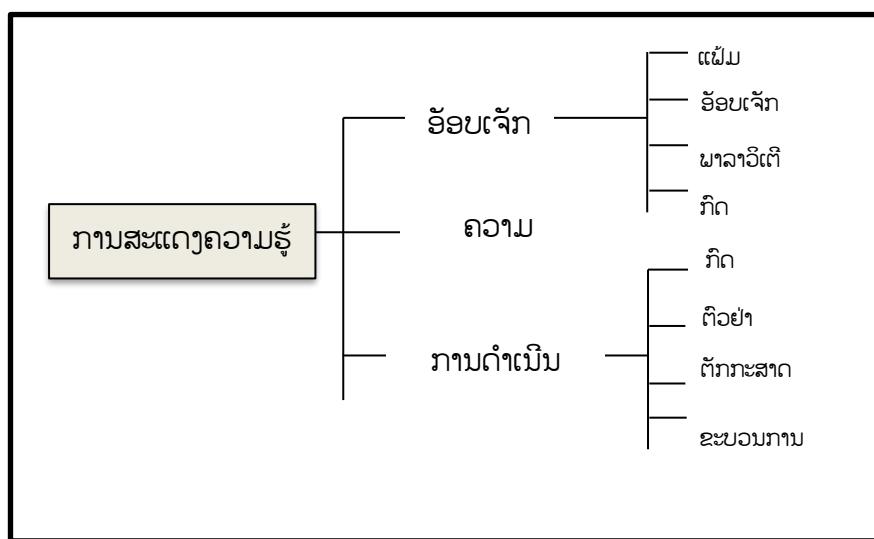
ເງື່ອນໄຂທຳອິດທີ່ໃຊ້ສໍາລັບພິຈາລະນາໃນການເລືອກ ESBT ຕີວິທີການໃນການສະແດງຄວາມຮູ້ຈະຕ້ອງງ່າຍ. ຈາກຮູບທີ່ 8.5 ຈະເຫັນວ່າການສະແດງຄວາມຮູ້ປະກອບດ້ວຍ 3 ຢ່າງຄື: Object Descriptions, Certainties ແລະ Actions. 3 ຢ່າງນີ້ເປັນຫົວຂໍ້ພື້ນຖານຂອງການສະແດງຄວາມຮູ້ ທີ່ຈະຕ້ອງມີເພື່ອປະກອບກັນເປັນຖານຄວາມຮູ້ໃນຖານຄວາມຮູ້ທີ່ເປັນແບບກົດ. ອອບເຈັກຈະຖືກສະແດງໄວ້ໃນສ່ວນຂອງຫຼັງ if ແລະ Action ຈະຖືກສະແດງໄວ້ໃນສ່ວນຂອງຫຼັງ then ສໍາລັບ Certainties ແມ່ນສ່ວນທີ່ເປັນ cf.

Object Descriptions เป็นส่วนที่ก่อภารกิจของห้องเรียนที่ใช้ในท่านความรู้, วิธีการสอนและความรู้ที่ใช้ที่ว่าไปเมื่อเยื่อ. โดยปีภาคติแล้วการสอนและความรู้แบบนี้จะสามารถตอบคุณเต็มที่ของการสอนที่มีความต้องการของผู้เรียน

ທີ່ມີວິທີການສະແດງຄວາມຮູ້ທີ່ດີແລ້ວໃນຖານຄວາມຮູ້ໜຶ່ງຈະສາມາດມີ Multiple World ໄດ້ໄດຍການສະແດງຄວາມຮູ້ທີ່ແບ່ງອອກເປັນເຝັມຫຼາຍອັນ.

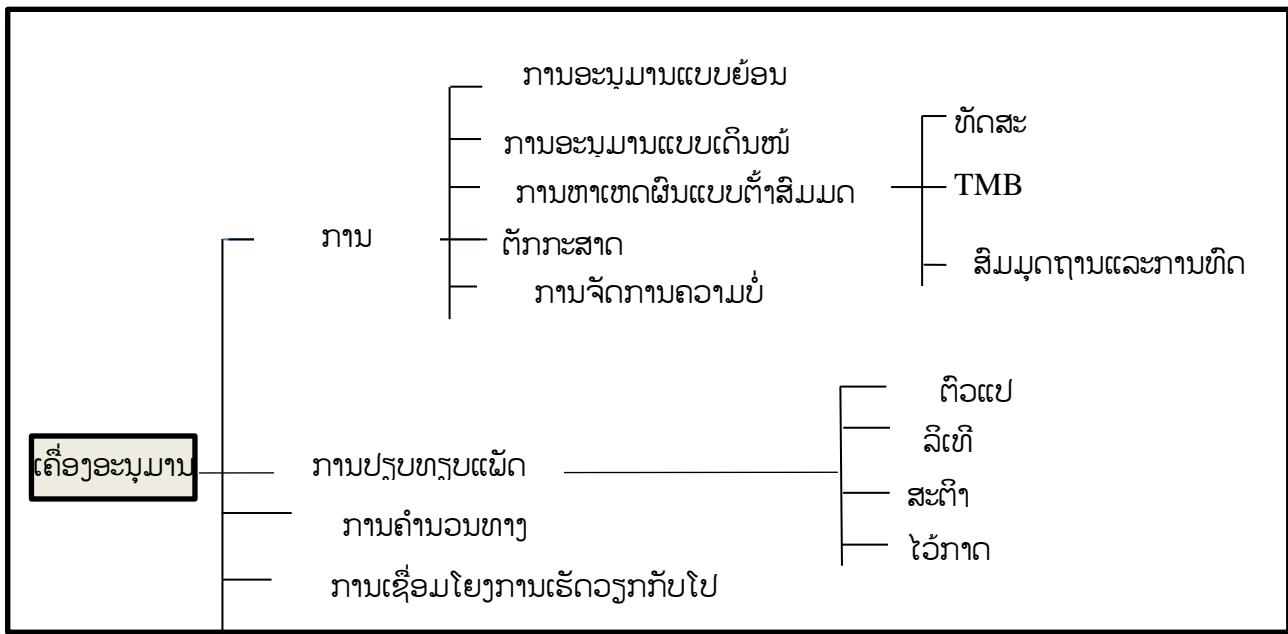
ການດຳເນີນການ (Actions) ເປັນສ່ວນທີ່ປ່ຽນສະຖານະການ ຫຼືຖານຂໍ້ມູນທີ່ກ່ຽວຂ້ອງ. ໃນສ່ວນຂອງການດຳເນີນໃນຖານຄວາມຮູ້ນັ້ນສາມາດສະແດງໄດ້ຫຼາຍວິທີ, ແຕ່ໄດຍສ່ວນໃຫ່ຍແລ້ວການດຳເນີນການຈະສະແດງຢູ່ໃນຮູບແບບຂອງກົດ ເຊິ່ງກົດເຫຼົ່ານີ້ອາດຈະຖືກນຳມາຈັດເປັນກຸ່ມເພື່ອຄວາມສະດວກໃນການປັບປຸງແກ້ໄຂ. ຮູບແບບໜຶ່ງທີ່ນີ້ມີໃຊ້ຫຼາຍກໍ່ວິທີການໃຊ້ຕົວຢ່າງ, ເຊິ່ງວິທີການສະແດງຄວາມຮູ້ເຊັ່ນນີ້ຈະງ່າຍກ່ວ່າແບບກົດຫຼາຍ.

ໃນການສະແດງຄວາມຮູ້, ໜີ້ທີ່ພັດທະນາລະບົບຜູ້ຊ່ວຂານຈະຕ້ອງແທນຄ່າຂອງອອບເຈັກ ແລະ ການດຳເນີນການປະກອບກັນເປັນຖານຄວາມຮູ້. ການແທນຄ່າອອບເຈັກນັ້ນ, ໜີ້ພັດທະນາລະບົບຈະຕ້ອງພິຈາລະນາເຖິງລະດົບຂອງຄວາມໝັ້ນ ໃຈ້ມີຕໍ່ຄວາມຮູ້ນັ້ນດ້ວຍ. ຖ້າຫາກໝັ້ນໃຈໃນຄວາມຮູ້ນັ້ນບໍ່ເຖິງ 100% ລັກສະນະຂອງການສະແດງຄວາມຮູ້ດັ່ງກ່າວຈະເອີ້ນວ່າ: ຄ່າຄວາມໝັ້ນໃຈ (Certainty Factor).



ຮູບທີ 8.5 ວິທີການຕ່າງໆ ຂອງການສະແດງ

❖ ເຕື່ອງອະນຸມານ



ຮບທີ 8.6 ການອະນຸມານຊະນິດຕ່າງໆ

❖ ການຕິດຕໍ່ກັບຜູ້ໃຊ້

ສຶ່ງທີ່ຈະຕັ້ງພິຈາລະນາໃນສ່ວນຂອງການຕິດຕໍ່ກັບຜູ້ໃຊ້ຄື:

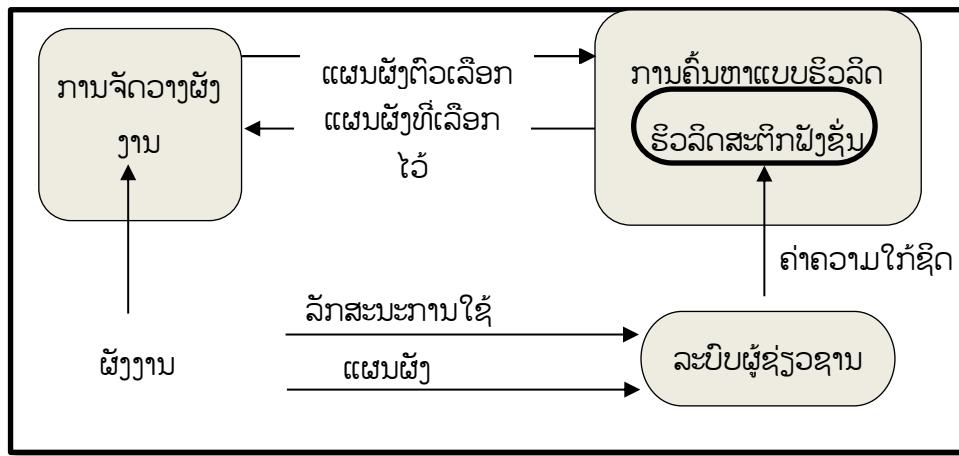
- វិធីការនៃការពេចចាប់តាម ហិច្ឆិដីខ្លួនដែលចាប់បញ្ចូលយ៉ាង
 - ការពេចចាប់តាមរាបតិន មើលច្ឆិដីខ្លួនខ្លួនទៅតាម, ផ្លូវ: why, how បែងពិនិត្យ
 - ការសម្រេចឱ្យជិនថ្មីនៅក្នុងការពេចចាប់តាម ដោយបង្ហាញពីការសម្រេចឱ្យរបស់វា, ផ្លូវ: ការសម្រេចឱ្យរបស់វា, ពាណិជ្ជកម្ម បែងពិនិត្យ
 - គោលរាល់នឹងការពេចចាប់តាម ដោយបង្ហាញពីការសម្រេចឱ្យរបស់វា, ផ្លូវ: ការសម្រេចឱ្យរបស់វា, ពាណិជ្ជកម្ម បែងពិនិត្យ

ທີ່ກ່າວໄປນັ້ນເປັນສ່ວນທີ່ມີຄວາມສໍາພັນໂດຍກົງກັບຜູ້ໃຊ້ລະບົບ ເພື່ອຮັດໃຫ້ຜູ້ໃຊ້ຄຸນເຕີຍກັບລະບົບ
ໄດ້ຈ່າຍ ແລະວ່ອງໄວ.

8.3. ການໃຊ້ລະບົບຜູ້ຂ່າວຊານສໍາລັບການວາງຜັງໂຮງງານ

ติวอย่างกามນำໃຊ้ລະບົບຜູ້ຂ່າວຊານສໍາລັບການວາງຜັງໂຮງງານມື້ງາຍ, ແຕ່ຕົວຢ່າງໃນທີ່ວິຊ້ນີ້ຈະເອົາຕົວຢ່າງທີ່ນຳສະເໜີໄດ້ຍຸນຈະເລີນ ແລະ ໄພລັດ (Sirinaovakul and Thajchayapong, 1994), ການເຮັດວຽກຂອງລະບົບສະແດງໃນຮູບທີ 8.7 ໄດຍເລີ່ມຕົ້ນຈາກຂະບວນການຈັດວາງເທື່ອລະພື້ນທີ່ເຮັດວຽກ ຄື: ເອົາພື້ນທີ່ເຮັດວຽກມາເຮັດການຈັດວາງເທື່ອລະພື້ນທີ່ ເພື່ອສ້າງເປັນແຜນຜັງຕົວເລືອກ (Layout Alternatives) ຫຼາຍຜັງໃຫ້ມື້ງາຍທີ່ສຸດເທົ່າທີ່ການຈັດວາງຈະເປັນໄປໄດ້. ໃນຕອນທຳອິດຈະເລີ່ມຕົ້ນຈາກ 2 ພື້ນທີ່ເຮັດວຽກ, ຈາກນັ້ນຂະບວນການຄົ້ນຫາແບບຮົວຮິດສະຕິກາ (heuristic) ຈະຄົ້ນຫາແຜນຜັງທີ່ດີຈາກແຜນຜັງຕົວເລືອກທັງໝົດຈະຖືກສ້າງຂຶ້ນ, ແລ້ວລະບົບຜູ້ຂ່າວຊານຈະເຮັດໜ້າທີ່ຂ່ວຍຂະບວນການການຄົ້ນຫາຮົວຮິດສະຕິກາໃນການເລືອກແຜນຜັງທີ່ດີດ້ວຍການກຳນົດຄ່າຄວາມສໍາພັນ (w) ໃຫ້ກັບຮົວຮິສຕິກາຜົງຊັນ ເມື່ອເລືອກພື້ນທີ່ທີ່ມີການຈັດວາງທີ່ດີທີ່ສຸດຂອງ 2 ພື້ນທີ່ໄດ້ແລ້ວ, ພື້ນທີ່ທີ່ 3 ຈະຖືກເອົາເຂົ້າ

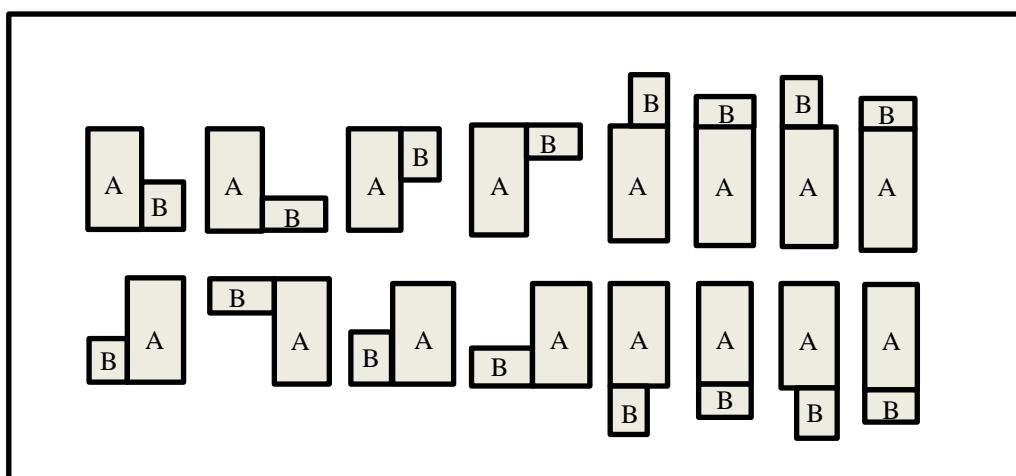
ມາຈັດວາງເພີ່ມເຂົ້າໄປໃນແຜນຜັງທີ່ໄດ້ເລືອກໄວ້ແລ້ວ ເພື່ອສ້າງແຜນຜັງຕົວເລືອກອອກມາອີກຈຳນວນນີ້ງ. ເຮັດການຂັດເລືອກດ້ວຍຂະບວນການຄົ້ນຫາແບບຮົດຮົດສົກອີກ ເມື່ອໄດ້ແຜນຜັງທີ່ດີອອກມາແລ້ວກໍໄສ໌ພື້ນທີ່ທີ 4 ເຂົ້າໄປອີກແລ້ວ, ເລືອກໃໝ່ຂະບວນການຂອງການເຮັດວຽກທັງລະບົບຈະໝູນວຸນເຊັ່ນນີ້ໄປເລືອຍໆຈົນບໍ່ມີພື້ນທີ່ໃໝ່ໃຫ້ວາງອີກ. ແຜນຜັງທີ່ດີທີ່ສຸດອັນສຸດທ້າຍກໍລິຜົນໄດ້ຮັບຂອງການວາງພື້ນທີ່ເຮັດວຽກແຜນພາບການເຮັດວຽກຂອງແຕ່ລະສ່ວນຈະສະແດງໃນຮູບທີ 8.7



ຮູບທີ 8.7 ແຜນຜັງການເຮັດວຽກຂອງລະບົບທີ່ນຳສະເໜີ

8.3.1 ການຈັດວາງພື້ນທີ່ເຮັດວຽກ

ການຈັດວາງພື້ນທີ່ເຮັດວຽກຈະເລີ່ມຈາກພື້ນທີ່ເຮັດວຽກຄູ່ທຳອິດ ໂດຍເລືອກຈາກພື້ນທີ່ເຮັດວຽກຄູ່ທີ່ມີຄ່າຄວາມສໍາພັນລະຫວ່າງພື້ນທີ່ເຮັດວຽກສູງສຸດຈາກຮູບທີ 8.8 ຕີ A ແລະ B ຈາກນັ້ນເອົາ B ໝູນຮອບຕົວເອງ, ແລ້ວຍ້າຍໄປຖຸກມູນຂອງ A ຈົນຄືບຫຼຸກດ້ານ. ໃນການໝູນໄປແຕ່ລະດ້ານຈະໄດ້ແຜນຜັງຕົວເລືອກອອກມານີ້ແບບເຮັດຊຳຂະບວນການເກົ່າເຊັ່ນນີ້ໄປເລືອຍໆຈົນ A ແລະ B ຊຶ່ງເອົາມາປະກອບກັນຫຼຸກດ້ານ.



ຮູບທີ 8.8 ການວາງພື້ນທີ່ເຮັດວຽກ A ແລະ B

ຈາກຮູບທີ 8.8 ຖ້າ A ແມ່ນເມເຈີ (Major) ແລະ B ແມ່ນໄມເນີ (Minor) ພື້ນທີ່ເຮັດວຽກ B ຈະໝູນຮອບໂຕເອງ, ແລ້ວໝູນຮອບພື້ນທີ່ເຮັດວຽກ A ໄດ້ແຜນຜັງອອກມາ 16 ແບບ. ໃນກໍລະນີທີ່ມີພື້ນທີ່ເຮັດວຽກ C ຖືກເລືອກເຂົ້າມາໃໝ່ ແລ້ວຈະຕັ້ງວາງລົງເທິງແຜນຜັງ AB ທີ່ໄດ້ຮັບເລືອກມາແລ້ວ ກໍໃຊ້ວິທີຄ້າຍກັນຄື: ໃຫ້ C ວາງຢູ່ຕໍ່ກັບດັນໄດ້ດັນໜຶ່ງຂອງພື້ນທີ່ເຮັດວຽກ A ຫຼື B ຈາກນັ້ນກໍເຮັດການໝູນ C ຮອບຕົວເອງ, ແລ້ວໝູນຮອບ B, ຈາກນັ້ນກໍໝູນຮອບ A ຈາກຜົນຂອງການໝູນທັງໝົດໃຫ້ເລືອກສະເພາະແຜນຜັງທີ່ບໍ່ມີພື້ນທີ່ເຮັດວຽກທີ່ວາງຊອນຫັບກັນແຜນຜັງທີ່ໄດ້ອອກມາຈະມີຫຼວງໝາຍ ຕາຕະລາງທີ 8.1 ເປັນຈຳນວນແຜນຜັງທີ່ໄດ້ກັບຈຳນວນພື້ນທີ່ເຮັດວຽກ.

ຕາຕະລາງ 8.1 ຈຳນວນແຜນຜັງທີ່ສ້າງຂຶ້ນຕໍ່ຈຳນວນພື້ນທີ່ເຮັດວຽກ

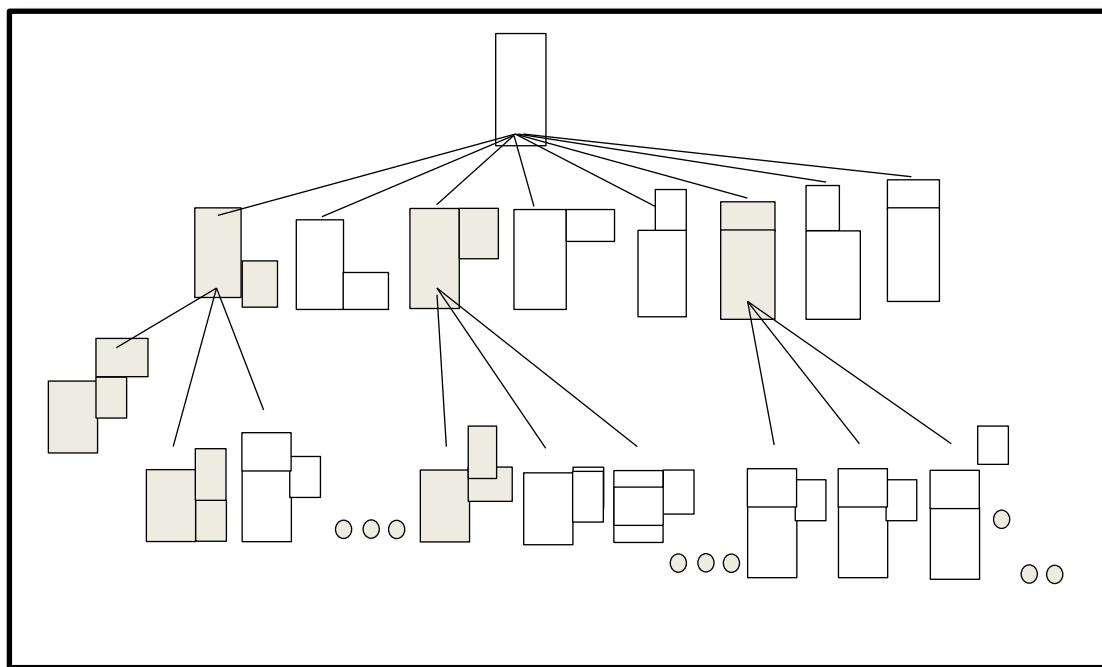
ຈຳນວນໜ່ວຍງານ	ຈຳນວນແຜນຜັງທີ່ໄດ້
2	32
3	1,792
4	143,360
5	14,909,440
6	1,908,408,320

8.3.2 ການຄົ້ນຫາແບບຮືບຮືດສະຕິກາ (Heuristic Search)

ການຄົ້ນຫາຈະໃຊ້ການຄົ້ນຫາແບບບິມ (Beam Search) ໂດຍເລີ່ມຕົ້ນຈາກການຄັດເລືອກພື້ນທີ່ເຮັດວຽກທີ່ມີຄ່າຄວາມສໍາພັນລະຫວ່າງພື້ນທີ່ເຮັດວຽກສູງສຸດ, ແລ້ວໃຊ້ອານຸກໍລິທຶນ (algorithm) ການຈັດວາງຮູບສ້າງແຜນຜັງຕົວເລືອກພົມຄໍານວນຄ່າໃຊ້ຈ່າຍຂອງແຕ່ລະແຜນຜັງ ໂດຍໃຊ້ຮືບຮືດສະຕິກາພັ້ງຊັ້ນ (heuristic function) ຈາກຄ່າໃຊ້ຈ່າຍນີ້ແຜນຜັງທີ່ມີຄ່າໃຊ້ຈ່າຍນ້ອຍກໍຈະຖືກເລືອກອອກມາຈຳນວນໜຶ່ງ. ຈາກນັ້ນລະບົບກໍຈະເລືອກພື້ນທີ່ເຮັດວຽກ ໃໝ່ທີ່ຈະນຳເຂົ້າມາວາງຕໍ່ໄປ ໂດຍພິຈາລະນາຈາກຄ່າຄວາມສໍາພັນສູງສຸດພື້ນທີ່ເຮັດວຽກໃໝ່ຈະຖືກນຳມາວາງເທິງແຜນຜັງທີ່ເລືອກມາແລ້ວຈາກຂັ້ນຕອນທີ່ຜ່ານມາ ໂດຍອາໄສຂະບວນການຈັດວາງຮູບແບບຄໍານວນຄ່າໃຊ້ຈ່າຍຂອງແຕ່ລະແຜນຜັງເພື່ອນຳມາພິຈາລະນາໃນການເລືອກອີກ. ຂະບວນການຂອງການຄົ້ນຫານີ້ຈະເຮັດຊ້າເຊັ່ນນີ້ໄປເລື້ອຍໆ ຈົນກວ່າຫຼຸກພື້ນທີ່ເຮັດວຽກຖືກວາງທັງໝົດ ແຜນຜັງຊຸດສຸດທ້າຍທີ່ຖືກເລືອກອອກມາຈະເປັນແຜນຜັງທີ່ຕັ້ງການ. ສະຫຼຸບແລ້ວຂະບວນການວາງແຜນຜັງຂອງລະບົບທີ່ນຳສະເໜີຈະເປັນຕໍ່ງນີ້:

1. ວາງພື້ນທີ່ເຮັດວຽກໂດຍຂະບວນການວາງພື້ນທີ່ເຮັດວຽກຜົນທີ່ໄດ້ຄືແຜນຜັງຕົວເລືອກ (Alternative layouts)
2. ຈາກຜົນທີ່ໄດ້ຈາກຂັ້ນຕອນທີ 1, ເລືອກແຜນຜັງທີ່ມີຄ່າໃຊ້ຈ່າຍຕໍ່ອອກມາຈຳນວນໜຶ່ງ
3. ຖ້າຍັງມີພື້ນທີ່ເຮັດວຽກທີ່ຕັ້ງກາງ, ໃຫ້ເລືອກພື້ນທີ່ເຮັດວຽກໃໝ່ທີ່ຈະຈັດວາງແລ້ວກັບໄປເຮັດຂັ້ນຕອນທີ 1. ໃນຮູບທີ 8.9 ສະແດງຂັ້ນຕອນການເຮັດວຽກຕາມຂໍສະຫຼຸບຂ້າງຕົ້ນ, ໂດຍເລີ່ມຈາກສ່ວນເທິງສຸດຂອງໂຄງສ້າງຕົ້ນໄມ້ທີ່ໄດ້ຈາກການເລືອກພື້ນທີ່ເຮັດວຽກຄຸ້່ທີ່ມີຄວາມໄກ້ຂີດສູງສຸດແລ້ວ

ນໍາພື້ນທີ່ເຮັດວຽກທີ່ມີພື້ນທີ່ໝາຍສຸດເປັນພື້ນທີ່ເຮັດວຽກເລີ່ມຕົ້ນ. ຈາກນັ້ນກໍເຮັດການຈັດວາງພື້ນທີ່
ເຮັດວຽກທີ2 ຈະໄດ້ຮູບແບບຕ່າງໆຂອງການຈັດວາງພື້ນທີ່ເຮັດວຽກອອກມາໝາວ່ງໝາຍ (ໃນຮູບ
ສະແດງໄວ້ແຕ່8ແຜນຜັງເທົ່ານັ້ນ) ແລ້ວເຮັດການເລືອກແຜນຜັງທີ່ດີທີ່ສຸດອອກມາ 3 ຜັງ (ບົມວິດ=
3) ເຊິ່ງກໍຕື່ແຜນຜັງທີ່ແລງເງິນໄວ້ໃນລະດັບທີ 1 ຂອງໂຄງສ້າງຕົ້ນໄມ້ ແລະ ຈາກແຜນຜັງທີ່ເລືອກ
ໄວ້ທັງສາມກໍເຮັດການວາງພື້ນທີ່ເຮັດວຽກທີ3ໃນແຕ່ລະພື້ນທີ່ເຮັດວຽກທີ່ວາງໄວ້ຈະໄດ້ແຜນຜັງລູກ
ອອກມາອີກຈຳໝາຍ(ໃນທີ່ນີ້ສະແດງໄວ້ແຕ່3ແຜນຜັງ)ຕາມທີ່ສະແດງໄວ້ໃນລະດັບທີ2ຂອງໂຄງສ້າງ
ຕົ້ນໄມ້ແລະຈາກທັງໝົດໃຫ້ເລືອກແຜນຜັງທີ່ດີທີ່ສຸດອອກມາ3 ແຜນຜັງຕາມທີ່ແລງເງິນແລ້ວເຮັດການ
ໃສ່ພື້ນທີ່ໃຫ້ວາງອີກເຜີນການເລືອກແຜນຜັງເທິ່ງສຸດທ້າຍກໍຕື່ແຜນຜັງທີ່ເປັນເຜີນຮັບ.



ຮັບທີ 8.9 ຂັ້ນຕອນການຈັດວາງ ແລະ ການຄົ້ນຫາແບບຍືນ

8.3.3 ລະບົບຜູ້ຊ່າງວຊານ

ຖານຄວາມຮູ້ໃນລະບົບຜູ້ຊ່ວຊານຄືກົດທີ່ສະແດງຂະບວນການກຳນົດຄ່າຄວາມສຳພັນຂອງພື້ນທີ່ເຮັດວຽກ(ຫຼືສຳປະສິດຄວາມແຂງຕຶງ)ໂດຍພິຈາລະນາຈາກໄລຍະການໃຫ້ງານງານຂອງພື້ນທີ່ການເຮັດວຽກນັ້ນໆຖ້າຈະຕ້ອງວາງພື້ນທີ່ເຮັດວຽກໄວ້ຕິດກັນຄ່າຄວາມສຳພັນນີ້ຈະຕ້ອງມີຄ່າສູງຖ້າຈະຕ້ອງພື້ນທີ່ເຮັດວຽກຫ່າງກັນຄ່າສຳປະສິດນີ້ຈະຕ້ອງມີຄ່າຕໍ່ໃນບາງຄັ້ງການກຳນົດນີ້ອາດຈະຕ້ອງກຳນົດພື້ນເສດອອກໄປເຊັ່ນ:ໃນໂຮງງານທີ່ມີພື້ນທີ່ເຮັດວຽກເຊື່ອມແລະພື້ນທີ່ເຮັດວຽກທຳຄວາມສະອາດຜົວວັດສະດຸເມື່ອພິຈາລະນາຢ່າງຄ່າວ່າແລວພື້ນທີ່ເຮັດວຽກທັງສອງນີ້ຄວນຢູ່ຕິດກັນເພາະເຊື່ອມແລວຈະໄດ້ເຮັດຄວາມສະອາດເລີຍແຕ່ໃນການວາງແຜນຈິງອາດເຮັດເຊັ່ນນີ້ບໍ່ໄດ້ເພາະພື້ນທີ່ສຳລັບເຮັດຄວາມສະອາດຜົວວັດ

LABEL: IF CONITION

IHEN WEIGHT=VALUE.

ໃນកົດຂອງIF_THANເມືອLABELເປັນຊີ່ກຳກັບຂອງກົດCONDITIONສໍາພັນຂອງພື້ນທີ່ເຮັດວຽກWEIGHTເປັນຄ່າສະເພາະທີ່ຈະຕ້ອງມີສະເໜີໃນກົດທຸກຂຶ້ນແລະVALUEເປັນຄ່າທີ່ຜູ້ຂ່າງວິການດີພາຍໃຕ້ເງື່ອນໄຂຂອງCONDITION

ពិវឌ្ឍន៍រាជការនៃជាតុកខេត្តខេត្តទាំងមួយ

ໃນໂຮງງານຂະណາດນ້ອຍແທ່ງໜຶ່ງມີພື້ນທີ່ເຮັດວຽກທີ່ເຮັດໜ້າທີ່ຕ່າງໆດັ່ງຕໍ່ໄປນີ້

1. ຫ້ອງຂຶ້ນຮູບໄລຍະ
 2. ຫ້ອງເຊື່ອມ
 3. ຫ້ອງເຮັດຄວາມສະອາດດ້ວຍສານເຄີມ
 4. ຫ້ອງເກັບເຕື້ອງ

ຈາກຂໍ້ມູນເກົ່າທີ່ເຕີຍເຮັດການບັນທຶກໄວ້ງ່າງວັກຂໍ້ມູນຕ່າງໆຂອງການໄຫຼຂອງວັດສະດຸໄປຕາມ
ຫ້ອງແລະຂະນາດຫ້ອງເປັນດັ່ງນີ້

ຕາຕະລາງທີ 8.2 ຂໍ້ມູນຂອງພື້ນທີ່ເຮັດວຽກຕ່າງໆ

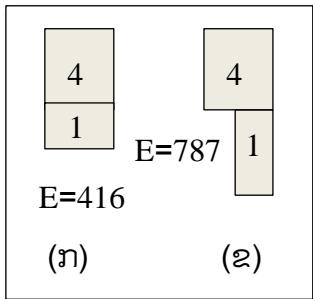
	ຫ້ອງຂຶ້ນຮູບ	ຫ້ອງເຊື່ອມ	ຫ້ອງທຳຄວາມສະອາດ	ຫ້ອງເກັບເຄື່ອງ	ຂະໜາດ
ຫ້ອງຂຶ້ນຮູບ 1	-	20	10	20	2x1
ຫ້ອງເຊື່ອມ 3	-	-	20	18	2x1
ຫ້ອງທຳຄວາມສະອາດ 3	-	-	-	15	1x1
ຫ້ອງເກັບເຄື່ອງ 4	-	-	-	-	2x2

ລັກສະນະຂອງຫ້ອງຕ່າງໆເປັນດັ່ງນີ້ຫ້ອງເຮັດຄວາມສະຫະອາດແລະຫ້ອງເຊື້ອມຈະຢູ່ຕິດກັນບໍ່ໄດ້
ວັດສະດຸໃຫ້ວູນລະຫ່ວາງຫ້ອງຂຶ້ນຮູບ,ຫ້ອງເຊື້ອມ,ຫ້ອງເກັບຂອງແລະຫ້ອງເຮັດຄວາມສະອາດມີ
ລັກສະນະໜັກຫຼາຍແລະເສຍຫາຍງ່າຍສຳລັບວັດສະດຸທີ່ໃຫ້ວູນລະຫ່ວາງຫ້ອງເຊື້ອມແລະຫ້ອງເຮັດຄວາມ
ສະອາດສົ່ງຕໍ່ຄວາມເສຍຫາຍງ່າຍ

ໂດຍການໃຫ້ຄໍາປຶກສາຂອງລະບົບຜູ້ຂ່າວຊານຈະໄດ້ຄໍາຄວາມສໍາພັນເປັນດັ່ງນີ້

W=5.1, w=8.97, w= 9.26, w=0,w= 3 ແລະ w=8.53

ຄ່າທີ່ໝາຍທີ່ສຸດຂອງພ ຕື ວ14ດັ່ງນັ້ນໃຫ້ເລືອກພື້ນທີ່ເຮັດວຽກ1ແລະ4ມາເຮັດການຈັດວາງແລະເລືອກແຜນຜັງທີ່ດີທີ່ສຸດອອກມາ2ຜັງດັ່ງສະແດງໃນຮູບທີ 8.10



ຮູບທີ 8.10 ຜົນຮັບຂອງການວາງພື້ນທີ່ເຮັດວຽກ 1 ແລະ 4

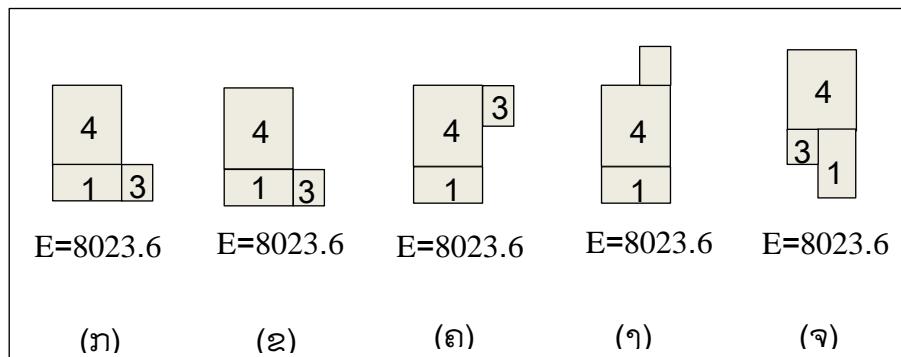
งานเลือกพื้นที่รีดวูกรหัส化ของตัวบัญชาและนาดึงนี้

- ຄ່າລວມຂອງ w ຂອງພື້ນທີ່ເຮັດວຽກ 2 ກັບພື້ນທີ່ເຮັດວຽກທີ່ຈຳວາງແລ້ວໄດ້ແກ່ພື້ນທີ່ເຮັດວຽກ 2 ກັບພື້ນທີ່ເຮັດວຽກ 1 ແລະພື້ນທີ່ເຮັດວຽກ 2 ກັບພື້ນທີ່ເຮັດວຽກ 4 ເຖິງກັບ $w+w=5.1+3=8.1$

-ຄ່າລວມຂອງພຂອອັນທີເຮດວຽກ 3 ກັບພື້ນທີ່ຈຳລວາງແລວໄດ້ແກ່ພື້ນທີ່ເຮດວຽກ 3 ກັບພື້ນທີ່ເຮດວຽກ 1 ແລະພື້ນທີ່ເຮດວຽກ 3 ກັບພື້ນທີ່ເຮດວຽກ 4 ເຖິງກັບ $w+w=8.97+8.53=112.5$

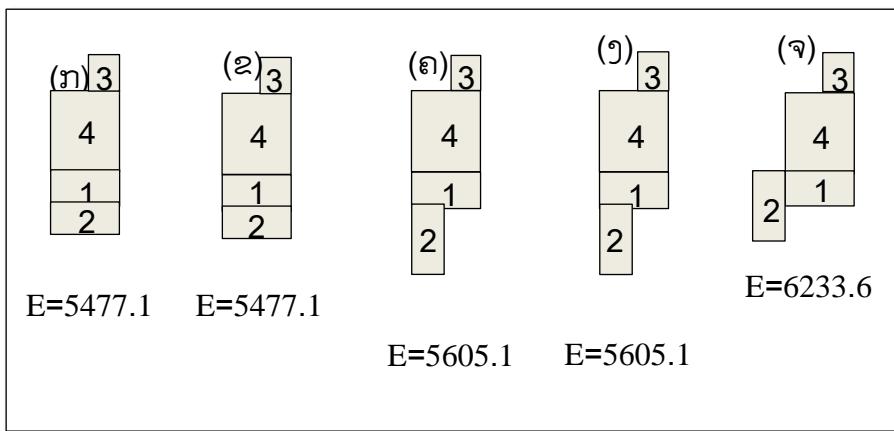
ຄ່າຄວາມສຳຄັນລວມຂອງພື້ນທີເຮັດວຽກ 3 ສູງກ່ວາຂອງພື້ນທີເຮັດວຽກ 2 ດັ່ງນັ້ນບໍ່ອນເຮັດ 3 ຜົນທີເຮັດຕໍ່ໄປທີຈະຈັດວາງເຊື່ອງເປີນຕາມຮູບທີ 8.11

3



ຮູບທີ 8.11ເຜີນຮັບຂອງການເລືອກແຜນຜັງໃສ່ພື້ນທີ່ເຮັດວຽກທີ 3

ຈາກນັ້ນໄສ່ພື້ນທີເຮັດວຽກທີ2ເຂົ້າໄປລັບຊື້ໄດ້ຕາມຮູບ 8.12



ຮູບທີ 8.12 ຜົນຮັບຂອງການວາງໃສ່ພື້ນທີ່ເຮັດວຽກ

ຜົນຂອງການວາງພື້ນທີ່ເຮັດວຽກຄືຮູບທີ 8.12 (ກ) ຫຼື (ຂ) ເຊິ່ງເປັນພຫັນທີ່ເຮັດວຽກທີ່ມີຄ່າ E ຕໍ່າທີ່ສຸດ ແລະ ຖື່ວ່າການເຮັດພື້ນທີ່ເຮັດວຽກທີ່ເປັນຄໍາຕອບ

ເອກະສານອ້າງອີງ

1. Mitchell, T. M. (1997). *Machine Learning*. New York: McGraw-Hill.
2. N.Sebe, Ira Chen, Ashtosh Garg and Thomas .S Huang. (2005). *Machine Learning in Computer Vision*. Dordrecht: Springer.
3. Norris, D. J. (2017). *Begining Artificial Intelligence with the Resbery Pi*. Barrington: Apress.
4. Rich, E. a. (1991). *Artificial Intelligence*. New York: McGraw-Hill.
5. Urwin, R. (2016). *Artificial Intelligence*. London: ARCTURUS.
6. ກິດຕິກຸນ, ບ. (2005). *Artificial Intelligence*. ກຸງເທບ.
7. ສີຮີເນົາວະກຸນ, ບ. (2014). *ບັນຍາປະເທດ*. ກຸງເທບ: ສໍານັກພິມທອບ.
8. ພັກດີວັດທະນະກຸນ, ກ. (2009). *ຄຳພືລະບົບສະໜັບສະໜູນການຕັດສິນໃຈ ແລະ ລະບົງໄຟຊົງວຂານ*. ກຸງເທບ: ເຄີຍ.