Chapter 5

Combining Intentions and Know-How

In the preceding chapters, I developed formalizations of intentions and know-how in a single general framework of actions and time. It is only fair to ask if these formalizations can be technically related in a useful manner. Indeed, they should be and can be. Putting them together makes it possible for us to derive results that are not derivable in other theories. As remarked in section 1.2, we need to be able to capture at least the following intuition concerning intentions and know-how: if an agent intends to achieve something and knows how to achieve it, then he will in fact bring it about.

This intuition can, in essence, be captured in the formal theory as developed so far, though some further technical restrictions need to be imposed to ensure that the given agent performs the appropriate actions. Note that this is intuitively much more reasonable than requiring, as Cohen & Levesque do [1990, p. 233], that if an agent intends to achieve something fanatically, he will succeed in bringing it about. Fanaticism is no cure for incompetence.

5.1 Some Basic Technical Results

In many ways, the definitions of intentions, ability, and know-how given in the preceding chapters paralleled each other. Their relationship to strategies and the progressive unraveling of strategies that those definitions involved were intuitively helpful factors in making their similarities obvious. I now consider some further results on the operators defined previously that help explicate their properties and are also helpful later on.

But, first, I define the formal language of this chapter, $\mathcal{L}^c$, by combining the rules for $\mathcal{L}^i$ and $\mathcal{L}^h$. Formally, $\mathcal{L}^c$ is defined as follows.

SYN-29. All the rules for $\mathcal{L}^i$, $\mathcal{L}^i_s$, $\mathcal{L}^i_y$, $\mathcal{L}^h$, $\mathcal{L}^h_s$, and $\mathcal{L}^h_y$, with $\mathcal{L}^c$ substituted for $\mathcal{L}^i$ and $\mathcal{L}^h$, $\mathcal{L}^c_s$ substituted for $\mathcal{L}^i_s$ and $\mathcal{L}^h_s$, and $\mathcal{L}^c_y$ substituted for $\mathcal{L}^i_y$ and $\mathcal{L}^h_y$

Lemma reflem-h-entails-i states the obvious result that if an agent knowingly performs $\mathbf{do}(q)$ on a scenario, then he performs $\mathbf{do}(q)$ on that scenario. Of course, if p holds at the end of the period over which he knowingly performs $\mathbf{do}(q)$, then p holds at the end of the period over which he performs $\mathbf{do}(q)$: indeed, they are the same period. This is because semantic definition 4.6 of $\langle\,\rangle_h$ requires that x know that q holds at the first moment at which it holds.

Lemma 5.1 $M \models_{S,t} x\langle \mathbf{do}(q)\rangle_h p$ implies $M \models_{S,t} x\langle \mathbf{do}(q)\rangle_i p$

Proof. $x\langle \mathbf{do}(q)\rangle_h p$ holds on a scenario, S, at a moment t iff there is a tree, τ, such that $(\exists t' \in S : [S;t,t'] \in [\![\tau]\!]^x_{\mathbf{do}(q)}$ and $M \models_{S,t'} p)$. The desired proof is by induction on the structure of trees. If τ is the empty tree, then $t = t'$; hence, $x\langle \mathbf{do}(q)\rangle_i p$ holds at S and t. Therefore, $[S;t,t] \in [\![\mathbf{do}(q)]\!]^x$. If τ is a single action, a, then t' is the moment of the first occurrence of q in the execution of a. Thus, $[S;t,t'] \in [\![\mathbf{do}(q)]\!]^x$. And, p holds at t'. Consequently, $M \models_{S,t} x\langle \mathbf{do}(q)\rangle_i p$. For a general tree, $\tau = \langle a;\tau_1,\ldots,\tau_m\rangle$, $[S;t,t'] \in [\![\tau]\!]^x_{\mathbf{do}(q)}$ iff $(\exists t_1,t_2,i : [S;t,t_1] \in [\![a]\!]$ and $[S;t_1,t_2] \in [\![\tau_i]\!]^x_{\mathbf{do}(q)}$ and t' is the first occurrence of q between t and $t_2)$. If a is sufficient to force q, i.e., if $t < t' \leq t_1$, then it is subsumed by the previous case. Otherwise, $[S;t_1,t'] \in [\![\tau_i]\!]^x_{\mathbf{do}(q)}$ holds. Assume, as the inductive hypothesis, that $[S;t_1,t'] \in [\![\mathbf{do}(q)]\!]^x$. Then, since there is no occurrence of q in $[S;t,t_1]$, we have $[S;t,t'] \in [\![\mathbf{do}(q)]\!]^x$. Thus $x\langle \mathbf{do}(q)\rangle_i p$ holds at t. □

It is important to note that $x\langle\!\langle Y\rangle\!\rangle p$ does not entail $x[\![Y]\!]p$, because x may not know how to follow strategy Y. For example, every action that leads to p may also occur on a scenario on which p never occurs. More surprisingly, $x[\![Y]\!]p$ does not entail $x\langle\!\langle Y\rangle\!\rangle p$. This is because, p might hold only on those scenarios on which the agent can force Y; it might not occur on other scenarios on which Y is (perhaps accidentally) performed. However, $x[\![Y]\!]p$ would imply $x\langle\!\langle Y\rangle\!\rangle p$ at those moments are considered from which either Y is not accidentally performed or its accidental performance does not lead to p. But, the agent may not have strategy Y anyway, i.e., $X * Y$ may be false, at even those moments.

In other words, we can show that intentions do not entail know-how, and know-how does not entail intentions. However, we do have the following positive result, which states that if (a) the agent can knowingly perform Y and (b) p holds at all moments at which Y is successfully performed, then the agent knows how to achieve p by performing Y.

Lemma 5.2 $\mathsf{A}(x\langle Y\rangle_i\mathsf{true}\rightarrow x\langle Y\rangle_i p) \wedge x[\!\langle Y\rangle\!]\mathsf{true}$ entails $x[\!\langle Y\rangle\!]p$

Proof. By SEM-48, $x[\!\langle Y\rangle\!]\mathsf{true}$ means that x knows that he can force $\downarrow_t Y$ and at each resulting moment, $x[\!\langle \uparrow_t Y\rangle\!]\mathsf{true}$ holds. By finitariness, this bottoms out after a finite number of recursive applications of SEM-48. Let t' be a moment at which that happens. Let $S \in \mathbf{S}_t$ be the scenario to which t' belongs. Then $M \models_{S,t} x\langle Y\rangle_i p$ holds. That is, p holds on t'. This is the case for all such moments t'. Therefore, $x[\!\langle Y\rangle\!]p$ holds at t. □

Note that $\mathsf{A}(x\langle Y\rangle_i\mathsf{true}\rightarrow x\langle Y\rangle_i p)$ is stronger than $x\langle\!\langle Y\rangle\!\rangle p$, which was defined as $\mathsf{A}(x\langle Y\rangle_i\mathsf{true}\rightarrow \mathsf{F}p)$ in Chapter 3.

5.2 Success at Last

The definition of know-how ensures that, if an agent knows how to achieve some condition, then on all scenarios on which he exercises his know-how, he will succeed in achieving the given condition. It might seem that nothing more remains to be said. However, many of our intuitions are about what conditions agents, in fact, bring about by performing their actions. The notion of "in fact" is captured in the model by the component **R**, which determines the real scenario at each moment.

Agents, at least those who are sufficiently rational, perform actions in order to achieve their intentions. We thus need to consider the actions an agent performs in trying to follow his strategy, since it is his strategy that determines his intentions at a given moment. This is intuitively quite natural: for an agent to succeed, he must have intentions that are commensurate with his know-how and must act so as to exercise his know-how. For example, an agent who intends to cross a river by swimming across it cannot be guaranteed to succeed if he walks away from the bank, i.e., if he does not act on his intention. He would also not be guaranteed to succeed in swimming across the river if the only way he knows how to cross the river is by walking across a bridge, i.e., if he lacks the know-how to swim across the river.

An agent's actual choices depend both on his beliefs and his strategies (which determine his intentions). For this reason, it is useful to define the notion of a strategy causing an action to be selected. A strategy of the form

do(q) *selects* an action if the agent knows that he can perform that action starting at the given moment and that on all scenarios on which that action is performed, either (a) the strategy will be successfully completed, or (b) he will know how to achieve it. In other words, a strategy selects an action if that action is the radix of a tree by following which the agent knows how to achieve the given strategy. I extend the notation of Chapter 4 so that $x\lfloor Y \rfloor a$ is also interpreted to mean that strategy Y selects action a for agent x. Formally,

SYN-30. $p \in \mathcal{L}^c_s$, $Y \in \mathcal{L}^c_y$, $a \in \mathcal{B}$, and $x \in \mathcal{A}$ implies that $(x\lfloor Y \rfloor a) \in \mathcal{L}^c$

I now give the semantic definition of $x\lfloor Y \rfloor a$ in two parts.

SEM-51. $M \models_t x\lfloor \mathbf{do}(q) \rfloor a$ iff $(\exists S, t', \tau_1, \ldots, \tau_m : [S; t, t'] \in [\![\{\langle a; \tau_1, \ldots, \tau_m \rangle\}]\!]^x_{\mathbf{do}(q)})$ and $M \models_t \neg q$

SEM-52. $M \models_t x\lfloor Y \rfloor a$ iff $\downarrow_t Y \neq \mathbf{skip}$ and $M \models_t x\lfloor \downarrow_t Y \rfloor a$

Since the above are all the cases in the definition of $x\lfloor Y \rfloor a$, an obvious consequence is that $M \not\models_t x\lfloor \mathbf{skip} \rfloor a$. This is only reasonable: the strategy **skip** does not call upon the agent to perform any actions at all. Indeed, actions may cause spurious changes in the state potentially affecting the executability of the strategy the agent might adopt next. The above definition is well-formed since, when $Y = \mathbf{do}(q)$, $Y = \downarrow_t Y$. This is the only possibility in which both cases apply.

By the definition of $[\![\{\ \}]\!]$, $x\lfloor \mathbf{do}(q) \rfloor a$ entails that a is doable on some scenario at the given moment. More importantly, a is selected by an agent's strategy only if the agent knows that it is an available and safe choice for achieving the given strategy. In other words, a is selected only if the agent knows that for any outcome of performing a, he will know how to achieve q, i.e., to select his next action, and so on.

We now have the requisite definitions in place to formally state that an agent will in fact perform an action that has been selected by his strategy. This, I-CONS-12, is the action selection constraint that was promised in constraint I-CONS-4 of section 3.3. Constraint I-CONS-12 states that if an agent has a strategy and some action is selected by that strategy, then the agent performs one of the actions selected by that strategy. The fact that some action is selected by a strategy means that the agent has the requisite know-how. Of course, a strategy may select more than one action: there may be several ways to achieve a given strategy. Therefore, all that is required is that the agent perform *one* of the selected actions.

I-CONS-12. **Selecting a sure action:**

$$(x * Y \wedge (\bigvee a : x\lfloor Y\rfloor a)) \rightarrow (\bigvee a : x\lfloor Y\rfloor a \wedge \mathsf{R}x\langle a\rangle \mathsf{true})$$

The above constraint is stronger than the one Newell calls the "principle of rationality" [1982, p. 102]. That principle merely requires that an agent select an action that he knows will lead to one of his goals; constraint I-CONS-12, by contrast, requires that an agent select an action only when he can force the success of his strategy by performing that action. I shall assume that this constraint applies throughout the following discussion. In conjunction with the persistence condition formalized in constraint I-CONS-5 of section 3.3, this leads to the following immediate consequences. The first is Lemma 5.3, which states that if an agent has a strategy and knows how to follow it, then he, in fact, succeeds in forcing the $\downarrow$ part of it to be executed. The second consequence is Lemma 5.4, which states that, if an agent has a strategy and knows how to follow it, then he eventually follows the $\downarrow$ part of it and, on doing so, adopts as his strategy the $\uparrow$ of his original strategy.

Lemma 5.3 $(x * Y \wedge x\langle\!\langle Y\rangle\!\rangle \mathsf{true} \wedge x\lfloor Y\rfloor \mathbf{do}(q)) \rightarrow \mathsf{R}\langle \mathbf{do}(q)\rangle_h \mathsf{true}$

Proof. Using the definitions of $\langle\!\langle\ \rangle\!\rangle$ and $\lfloor\ \rfloor$, we can infer $x\langle\!\langle \mathbf{do}(q)\rangle\!\rangle \mathsf{true}$ from $x\langle\!\langle Y\rangle\!\rangle \mathsf{true} \wedge x\lfloor Y\rfloor \mathbf{do}(q)$. And, $x\langle\!\langle \mathbf{do}(q)\rangle\!\rangle \mathsf{true}$ entails that either q holds or it is the case that $(\bigvee a : x\lfloor \mathbf{do}(q)\rfloor a \wedge \mathsf{A}[a]x\langle\!\langle \mathbf{do}(q)\rangle\!\rangle \mathsf{true})$. By the definition of $\lfloor\ \rfloor$, q cannot hold at the given moment. Therefore, by I-CONS-12, one of the actions, a, such that $x\lfloor \mathbf{do}(q)\rfloor a \wedge \mathsf{A}[a]x\langle\!\langle \mathbf{do}(q)\rangle\!\rangle \mathsf{true}$, occurs on the real scenario. Since $x\langle\!\langle \mathbf{do}(q)\rangle\!\rangle \mathsf{true}$ holds iff there exists a tree with the appropriate properties, we can induce on the structure of trees to obtain the desired result. □

Lemma 5.4 $(x * Y \wedge x\langle\!\langle Y\rangle\!\rangle \mathsf{true} \wedge x\lfloor Y\rfloor \mathbf{do}(q)) \rightarrow \mathsf{R}\langle \mathbf{do}(q)\rangle_i (x * \uparrow_t Y)$

Proof. By Lemma 5.3, we have $\mathsf{R}\langle \mathbf{do}(q)\rangle_h \mathsf{true}$, which by Lemma 5.1 entails $\mathsf{R}\langle \mathbf{do}(q)\rangle_i \mathsf{true}$. From $x\lfloor Y\rfloor \mathbf{do}(q)$, we can infer that $\downarrow_t Y \neq \mathbf{skip}$ (in fact, $\downarrow_t Y = \mathbf{do}(q)$). Therefore, constraint I-CONS-5 of section 3.3 applies and we obtain $\mathsf{R}[\downarrow_t Y]_i (x * \uparrow_t Y)$, which is the same as $\mathsf{R}[\mathbf{do}(q)]_i (x * \uparrow_t Y)$. In the presence of $\mathsf{R}\langle \mathbf{do}(q)\rangle_i \mathsf{true}$, this yields the desired result. □

Lemma 5.5 $(x * Y \wedge x\langle\!\langle Y\rangle\!\rangle \mathsf{true}) \rightarrow \mathsf{R}\langle Y\rangle_i \mathsf{true}$

Proof. Consider a moment t at which the antecedent holds. If $\downarrow_t Y = \mathbf{skip}$, then $\mathsf{R}\langle \downarrow_t Y\rangle_i \mathsf{true}$ holds vacuously. If $\downarrow_t Y = \mathbf{do}(q)$, then $x\lfloor Y\rfloor \mathbf{do}(q)$ holds because of the semantic definition of $x\langle\!\langle Y\rangle\!\rangle \mathsf{true}$ (SEM-48). Hence, $\mathsf{R}\langle \downarrow_t Y\rangle_h \mathsf{true}$ holds by Lemma 5.3, which by Lemma 5.1 entails $\mathsf{R}\langle \downarrow_t Y\rangle_i \mathsf{true}$.

By Lemma 5.4, $\mathsf{R}\langle\downarrow_t Y\rangle_i(x{*}\uparrow_t Y)$. Also, by the definition of $x[\![\langle Y\rangle]\!]\mathsf{true}$ (i.e., SEM-48), we obtain that $\mathsf{R}\langle\downarrow_t Y\rangle_i x[\![\langle\uparrow_t Y\rangle]\!]\mathsf{true}$. Thus, we can apply mathematical induction on the depth of strategies. The depth of a strategy is defined, as before, as the number of recursive applications of $[\![\langle\ \rangle]\!]$ needed to evaluate $[\![\langle Y\rangle]\!]$. The strategy $\uparrow_t Y$ has a smaller depth than Y. The base cases of **skip** and **do**(q) were considered above. Therefore, by induction, we have $\mathsf{R}\langle\downarrow_t Y\rangle_i\mathsf{true}$. Using Lemma 3.5, we obtain the desired result. □

Theorem 5.6 $(x * Y \wedge x\langle\!\langle Y\rangle\!\rangle p \wedge x[\![\langle Y\rangle]\!]\mathsf{true}) \rightarrow \mathsf{RF}p$

Proof. From the antecedent of this claim and Lemma 5.5, we conclude that $\mathsf{R}\langle Y\rangle_i\mathsf{true}$ holds. By the semantic definition of $x\langle\!\langle Y\rangle\!\rangle p$, we obtain $\mathsf{RF}p$. □

Clearly, several actions may be selected by a given strategy: these are all the actions that are radices of trees with which that strategy may be achieved. All the actions that may be selected by the assigned strategy are treated on par. Thus only those conditions can be considered as forced by a given strategy that occur on all scenarios on which any of the selected actions is performed.

Constraint I-CONS-12 states that if an agent's strategy selects one or more actions, then he performs one such selected action. An important consequence of this is that if an agent knows how to execute his current strategy, then he performs some action to execute it. This prevents the kind of inaction that arises in Buridan's famous example. That example is of a donkey who cannot choose between two equally accessible and equally tempting bales of hay and thus starves to death. Assuming that only the actions of stepping towards one of the bales are selected by the donkey's strategy of obtaining food, constraint I-CONS-12 requires the donkey to choose one bale or the other.

5.3 Normal Models

The above definitions involve all possible scenarios. In particular, for an agent to know how to achieve p, he must have an action that limits the possible scenarios to those in which, perhaps by further actions, he can actually achieve p. Unfortunately, this may be too strong a requirement in real-life, because no action can be guaranteed to succeed. For example, I know how to drive to work, but can actually do so only if my car does not break down, and the bridge I drive over does not collapse, and so on. That is, there is a scenario over which the required condition will not occur. Thus success ought to be required, but only under *normal* conditions.

One way to improve the above definition is by only considering normal models. Explicit reasoning about whether a given model is normal, i.e., whether an agent has the ability to achieve some condition, however, involves nothing less than a solution to the qualification and ramification problems of McCarthy & Hayes [1969]. This is because to infer whether an agent has some ability, we would have to reason about when different basic actions would be doable by him (this is the qualification problem), and what effects they would have if done in a given state (this is the ramification problem). This issue is not addressed here.

This point was discussed in section 1.2 in relation to the distinction between model construction and usage. The definitions given here assume that a reasonable model has been constructed; nonmonotonic reasoning must be used when model construction itself is considered.

5.4 Other Theories of Ability and Know-How

I now briefly review the other formal theories of ability and know-how. I discuss them here rather than in Chapter 4, since it important to compare them with the results of this chapter, as well as those of Chapter 4.

Oddie & Tichy have proposed a nice theory of ability, which I became aware of only during the last stages of completing this manuscript [Oddie & Tichy, 1981; Tichy & Oddie, 1983]. Oddie & Tichy share many of the intuitions of the present approach, though their ultimate goal is to explicate the notions of ability and opportunity as those notions may be used in characterizing freedom and responsibility. They agree with the present approach in postulating branching models to capture the different choices that agents may make. They also agree in considering different possible consequences of an agent's actions in determining whether he has the ability to force something [1983, p. 135].

Oddie & Tichy consider strategies as in classical game theory. They define ability using strategies. They formally define strategies as *trees*, which for them are fragments of the model that include some possible futures [1983, p. 139]. Their trees are thus different from the trees of the present approach. However, their trees resemble, to some extent, the notions of ability-intension and know-how-intension, which were defined here. The key difference is that Oddie & Tichy's agents may not be able to ensure that the world evolves along one of the branches of their trees; by contrast, the definitions of ability-intension and know-how-intension are such that agents can always ensure that the world evolves according to one of their member periods.

After developing a logic of opportunity, which I shall not discuss, Oddie & Tichy define the concept of ability. I simplify their definitions slightly for ease of exposition. A *steadfast* intention is one that the agent will persist with until he succeeds. An agent is able to achieve A if there is a strategy that ensures A and which the agent *commands* with respect to A [1981, p. 243]. An agent commands a strategy with respect to a condition A if a steadfast intention on part of the agent to achieve A is sufficient to *heed* that strategy (i.e., to successfully follow it).

Thus, Oddie & Tichy's technical definition of ability is quite different from the one given above. They define ability in terms of intentions. However, intentions themselves are not formally defined. Constraints between intentions and actions or ability and actions are not stated in their theory. Paradigmatic examples of such constraints are (a) that agents act according to their intentions and (b) that they may exercise their abilities under certain circumstances. Indeed, if such constraints were stated directly, as they are in the present approach, there would be no need to define ability in terms of intentions. Despite these differences, Oddie & Tichy are able to prove a version of the success theorem, which states that ability conjoined with a steadfast intention implies performance [1983, p. 145]. Of course, this theorem would hold in general only in the presence of constraints such as the ones discussed in this chapter.

A theory of ability has also been proposed by Werner [1991]. Some other aspects of that paper were discussed in section 3.5. Werner assigns information states, I, to agents. He defines $\text{Alt}(I)$ as the alternatives or choices available to an agent with information I (p. 112). These are the actions that the agent may perform. This assignment seems counterintuitive in that one would expect the choices available to an agent to be independent of the information he has. Of course, how the agent actually exercises those choices would depend on his information. But that process would also depend on the agent's intentions at that time. Werner defines the intentional state of an agent, but only uses the agent's information state in determining the agent's choices: either both should be considered (to capture the options an agent is focusing on), or neither (to capture all physical options).

Strategies are defined as in game theory. A strategy, δ, is a function from information states to choices (p. 113). $\delta(I) \subseteq \text{Alt}(I)$. That is, the choices picked by a strategy are the ones available given the agent's information state. Ability is defined as follows: an agent *can* achieve p iff he has a strategy for p. A strategy is for p iff p is realized in all histories compatible with that strategy. The distinction between past, present, and future is never clearly delineated.

Thus agents may have the ability to achieve past conditions, just as well as future conditions. This seems problematic for any useful notion of ability.

Actions and time are not a part of Werner's formal language. Few logical inferences are given and there is no axiomatization. The only logical property of his definition of ability that Werner notes is the obvious claim that it lies somewhere between necessity and possibility (p. 114).

Independently of, and prior to, the present approach, Segerberg proposed a logic of achievements in the framework of dynamic logic [Segerberg, 1989]. Thus, in spirit, his work is similar to the theory developed here. He defines an operator δ, which takes a condition and converts it into an action, namely, the action of bringing about that condition. This is intuitively quite close to strategies of the form, $\mathbf{do}(q)$, as defined here. Segerberg uses actions of the form δq as the primitive actions in his variant of dynamic logic. This too is similar to the present approach: the only difference is that I have considered a deterministic version of dynamic logic.

However, there are some important dissimilarities. Segerberg gives the semantics of actions of the form δq in terms of all paths (i.e., computations) that result from any program α, such that α terminates only in states where q holds (p. 328). In the present approach, the corresponding notion is that of ability-intensions, which also end with an occurrence of the relevant condition. But there are two major differences. First, ability-intensions require that the given agent be able to *force* the relevant condition. Fortuitous occurrences of the condition and the paths over which they occur are simply eliminated from ability-intensions. Indeed, Segerberg's semantics seems to agree more with the definition of $[\![\,]\!]$ for strategies that was used to give a semantics of intentions in Chapter 3. Segerberg does not define intentions in his paper, but it is not clear how he would separate the concept of intentions from the concepts of ability and know-how. It is obvious, however, that we should not require that intentions entail ability or know-how.

Second, Segerberg does not consider the knowledge of agents. Thus the effects of agents' knowledge on their choices cannot be considered. Such choices arise in Segerberg's logic as tests on conditions and in the present approach in conditional and iterative strategies. Third, ability-intensions end at the first occurrence of the relevant condition, not an arbitrary one: this is important in considering executions of abstract strategies and in relating know-how with intentions, because it tells us just how far the current substrategy of a strategy will be executed before the rest of it kicks in. This is crucial for unambiguous definitions in the present approach.

Cohen & Levesque's success theorem [1990, p. 233] is one of the counterintuitive consequences of their theory. It allows an agent to succeed with an intention merely through persistence. The agent simply has to be able to correctly identify the intended condition; he does not need to know how to achieve it. Clearly, this requirement is not sufficient: one can come up with several natural conditions that an agent may be able to identify, but would not be able to achieve. A similar, and equally unsatisfactory, result is proved by Rao & Georgeff [1991a, section 4].

Both the abovementioned theories require an additional assumption to prove their respective success theorems. This is the assumption that agents will eventually drop their goals or intentions. If the dropping of a goal or intention is conditioned on the agent's obtaining certain true knowledge, success can easily be guaranteed. However, this is backwards from our pretheoretic intuitions. Agents may drop their intentions eventually, but we cannot force external events to occur on the basis of this change in internal state. In the present approach, an agent may drop an intention at any time, but success is not guaranteed unless he applies his know-how. Assuming that an agent will drop an intention upon success, it can be shown that an agent who meets the conditions of Theorem 5.6, will eventually drop his intention. Thus, under appropriate circumstances, a primitive assumption of the abovementioned theories becomes a consequence of the present approach.

I have shown how to formalize intentions and know-how and to combine them to characterize the behavior of an intelligent agent. However, the main goal of this monograph is to apply these abstractions to multiagent systems. This can be achieved in at least two ways. The agents' intentions, knowledge, and know-how can be used to succinctly describe their expected behavior and constrain it to capture various system properties. For instance, restrictions on how an agent may revise his intentions would be in this category. They can go a long way in specifying the behavior of an agent who is rational enough to act on his intentions. Similarly, constraints may be stated among the intentions, knowledge, and know-how of different agents in a multiagent system. This can, of course, be done directly in our formal language.

However, an extremely important class of interactions among agents pertains specifically to communications among them. The next chapter discusses how to formalize communications in a manner that focuses on their content, and not their form. It also provides a semantics for communications using intentions and know-how and shows how communication protocols can be specified in our approach.